GenESyS - GENeric and Expansible SYstem Simulator

v22.01 (jude)

Generated by Doxygen 1.9.1

1 Todo List 1

2 Namespace Index 1

2.1 Namespace List 1

3 Hierarchical Index 2

3.1 Class Hierarchy 2

4 Class Index 10

4.1 Class List 10

5 File Index 18

5.1 File List 18

6 Namespace Documentation 32

6.1 Ui Namespace Reference 32

6.2 yy Namespace Reference 32

6.2.1 Function Documentation 33

7 Class Documentation 36

7.1 Access Class Reference 36

7.1.1 Detailed Description 37

7.1.2 Constructor & Destructor Documentation 37

7.1.3 Member Function Documentation 38

7.2 AnElectronicAssemblyAndTestSystem Class Reference 39

7.2.1 Detailed Description 41

7.2.2 Constructor & Destructor Documentation 41

7.2.3 Member Function Documentation 41

7.3 Assign Class Reference 42

7.3.1 Detailed Description 43

7.3.2 Constructor & Destructor Documentation 43

7.3.3 Member Function Documentation 44

7.3.4 Member Data Documentation 45

7.4 Assign::Assignment Class Reference 46

7.4.1 Detailed Description 46

7.4.2 Constructor & Destructor Documentation 46

7.4.3 Member Function Documentation 46

7.4.4 Member Data Documentation 47

7.5 SequenceStep::Assignment Class Reference 47

7.5.1 Detailed Description 48

7.5.2 Constructor & Destructor Documentation 48

7.5.3 Member Function Documentation 48

7.5.4 Member Data Documentation 49

7.6 Attribute Class Reference 49

7.6.1 Detailed Description . . . 51
7.6.2 Constructor & Destructor Documentation . . . 51
7.6.3 Member Function Documentation . . . 52
7.7 Base Class Reference . . . 53
7.8 BaseConsoleGenesysApplication Class Reference . . . 54
7.8.1 Detailed Description . . . 54
7.8.2 Constructor & Destructor Documentation . . . 54
7.8.3 Member Function Documentation . . . 55
7.9 yy::genesyspp_parser::basic_symbol< Base > Struct Template Reference . . . 58
7.9.1 Detailed Description . . . 60
7.9.2 Member Typedef Documentation . . . 60
7.9.3 Constructor & Destructor Documentation . . . 60
7.9.4 Member Function Documentation . . . 61
7.9.5 Member Data Documentation . . . 63
7.10 Batch Class Reference . . . 63
7.10.1 Detailed Description . . . 65
7.10.2 Constructor & Destructor Documentation . . . 65
7.10.3 Member Function Documentation . . . 66
7.11 Book_Cap02_Example01 Class Reference . . . 67
7.11.1 Detailed Description . . . 69
7.11.2 Constructor & Destructor Documentation . . . 69
7.11.3 Member Function Documentation . . . 69
7.12 SamplerBoostImpl::BoostImplRNG_Parameters Struct Reference . . . 70
7.12.1 Detailed Description . . . 70
7.12.2 Member Data Documentation . . . 70
7.13 yy::genesyspp_parser::by_kind Struct Reference . . . 71
7.13.1 Detailed Description . . . 72
7.13.2 Member Typedef Documentation . . . 72
7.13.3 Constructor & Destructor Documentation . . . 72
7.13.4 Member Function Documentation . . . 73
7.13.5 Member Data Documentation . . . 74
7.14 yy::genesyspp_parser::by_state Struct Reference . . . 74
7.14.1 Detailed Description . . . 76
7.14.2 Member Typedef Documentation . . . 76
7.14.3 Member Enumeration Documentation . . . 76
7.14.4 Constructor & Destructor Documentation . . . 77
7.14.5 Member Function Documentation . . . 77
7.14.6 Member Data Documentation . . . 78
7.15 CelularAutomata Class Reference . . . 78
7.15.1 Detailed Description . . . 80
7.15.2 Constructor & Destructor Documentation . . . 80
7.15.3 Member Function Documentation . . . 80

7.16 Collector_if Class Reference . . . 82
7.16.1 Detailed Description . . . 84
7.16.2 Member Function Documentation . . . 84
7.17 CollectorDatafile_if Class Reference . . . 85
7.17.1 Detailed Description . . . 86
7.17.2 Member Function Documentation . . . 86
7.18 CollectorDatafileDefaultImpl1 Class Reference . . . 87
7.18.1 Detailed Description . . . 89
7.18.2 Constructor & Destructor Documentation . . . 89
7.18.3 Member Function Documentation . . . 89
7.18.4 Member Data Documentation . . . 91
7.19 CollectorDefaultImpl1 Class Reference . . . 92
7.19.1 Detailed Description . . . 93
7.19.2 Constructor & Destructor Documentation . . . 93
7.19.3 Member Function Documentation . . . 93
7.19.4 Member Data Documentation . . . 94
7.20 ComponentManager Class Reference . . . 95
7.20.1 Detailed Description . . . 96
7.20.2 Constructor & Destructor Documentation . . . 96
7.20.3 Member Function Documentation . . . 96
7.20.4 Member Data Documentation . . . 98
7.21 HypothesisTester_if::ConfidenceInterval Class Reference . . . 98
7.21.1 Detailed Description . . . 99
7.21.2 Constructor & Destructor Documentation . . . 99
7.21.3 Member Function Documentation . . . 99
7.21.4 Member Data Documentation . . . 99
7.22 ConnectionManager Class Reference . . . 100
7.22.1 Detailed Description . . . 100
7.22.2 Constructor & Destructor Documentation . . . 100
7.22.3 Member Function Documentation . . . 101
7.22.4 Member Data Documentation . . . 103
7.23 yy::genesyspp_parser::context Class Reference . . . 105
7.23.1 Detailed Description . . . 105
7.23.2 Constructor & Destructor Documentation . . . 105
7.23.3 Member Function Documentation . . . 105
7.23.4 Member Data Documentation . . . 106
7.24 Counter Class Reference . . . 107
7.24.1 Detailed Description . . . 108
7.24.2 Constructor & Destructor Documentation . . . 108
7.24.3 Member Function Documentation . . . 109
7.24.4 Member Data Documentation . . . 110
7.25 Create Class Reference . . . 111

7.25.1 Detailed Description . . . 112
7.25.2 Constructor & Destructor Documentation . . . 112
7.25.3 Member Function Documentation . . . 113
7.25.4 Member Data Documentation . . . 114
7.26 Decide Class Reference . . . 115
7.26.1 Detailed Description . . . 117
7.26.2 Constructor & Destructor Documentation . . . 118
7.26.3 Member Function Documentation . . . 118
7.26.4 Member Data Documentation . . . 120
7.27 Assign::DEFAULT_VALUES Struct Reference . . . 120
7.27.1 Detailed Description . . . 120
7.27.2 Member Data Documentation . . . 120
7.28 Delay::DEFAULT_VALUES Struct Reference . . . 121
7.28.1 Detailed Description . . . 121
7.28.2 Member Data Documentation . . . 121
7.29 EntityType::DEFAULT_VALUES Struct Reference . . . 121
7.29.1 Detailed Description . . . 121
7.29.2 Member Data Documentation . . . 121
7.30 ModelComponent::DEFAULT_VALUES Struct Reference . . . 122
7.30.1 Detailed Description . . . 122
7.30.2 Member Data Documentation . . . 122
7.31 ModelSimulation::DEFAULT_VALUES Struct Reference . . . 123
7.31.1 Detailed Description . . . 123
7.31.2 Member Data Documentation . . . 123
7.32 Queue::DEFAULT_VALUES Struct Reference . . . 125
7.32.1 Detailed Description . . . 125
7.32.2 Member Data Documentation . . . 126
7.33 QueueableItem::DEFAULT_VALUES Struct Reference . . . 126
7.33.1 Detailed Description . . . 126
7.33.2 Member Data Documentation . . . 126
7.34 Release::DEFAULT_VALUES Struct Reference . . . 127
7.34.1 Detailed Description . . . 127
7.34.2 Member Data Documentation . . . 127
7.35 Resource::DEFAULT_VALUES Struct Reference . . . 128
7.35.1 Detailed Description . . . 128
7.35.2 Member Data Documentation . . . 128
7.36 Route::DEFAULT_VALUES Struct Reference . . . 128
7.36.1 Detailed Description . . . 129
7.36.2 Member Data Documentation . . . 129
7.37 SeizableItem::DEFAULT_VALUES Struct Reference . . . 129
7.37.1 Detailed Description . . . 129
7.37.2 Member Data Documentation . . . 129

7.38 Seize::DEFAULT_VALUES Struct Reference . . . 130
7.38.1 Detailed Description . . . 130
7.38.2 Member Data Documentation . . . 131
7.39 SequenceStep::DEFAULT_VALUES Struct Reference . . . 131
7.39.1 Detailed Description . . . 131
7.39.2 Member Data Documentation . . . 131
7.40 Set::DEFAULT_VALUES Struct Reference . . . 132
7.40.1 Detailed Description . . . 132
7.40.2 Member Data Documentation . . . 132
7.41 SourceModelComponent::DEFAULT_VALUES Struct Reference . . . 132
7.41.1 Detailed Description . . . 133
7.41.2 Member Data Documentation . . . 133
7.42 Write::DEFAULT_VALUES Struct Reference . . . 134
7.42.1 Detailed Description . . . 134
7.42.2 Member Data Documentation . . . 134
7.43 WriteExpression::DEFAULT_VALUES Struct Reference . . . 134
7.43.1 Detailed Description . . . 134
7.43.2 Member Data Documentation . . . 135
7.44 WriteInExpression::DEFAULT_VALUES Struct Reference . . . 135
7.44.1 Detailed Description . . . 135
7.44.2 Member Data Documentation . . . 135
7.45 WriteInText::DEFAULT_VALUES Struct Reference . . . 136
7.45.1 Detailed Description . . . 136
7.45.2 Member Data Documentation . . . 136
7.46 WriteText::DEFAULT_VALUES Struct Reference . . . 137
7.46.1 Detailed Description . . . 137
7.46.2 Member Data Documentation . . . 137
7.47 SamplerDefaultImpl1::DefaultImpl1RNG_Parameters Struct Reference . . . 138
7.47.1 Detailed Description . . . 139
7.47.2 Constructor & Destructor Documentation . . . 139
7.47.3 Member Data Documentation . . . 139
7.48 Delay Class Reference . . . 140
7.48.1 Detailed Description . . . 141
7.48.2 Constructor & Destructor Documentation . . . 142
7.48.3 Member Function Documentation . . . 142
7.48.4 Member Data Documentation . . . 144
7.49 Ui::DialogAbout Class Reference . . . 145
7.49.1 Detailed Description . . . 146
7.50 Ui::DialogModelInformation Class Reference . . . 146
7.50.1 Detailed Description . . . 147
7.51 Dispose Class Reference . . . 147
7.51.1 Detailed Description . . . 149

7.51.2 Constructor & Destructor Documentation . . . 149
7.51.3 Member Function Documentation . . . 150
7.51.4 Member Data Documentation . . . 151
7.52 DropOff Class Reference . . . 152
7.52.1 Detailed Description . . . 153
7.52.2 Constructor & Destructor Documentation . . . 153
7.52.3 Member Function Documentation . . . 153
7.53 Dummy Class Reference . . . 155
7.53.1 Detailed Description . . . 157
7.53.2 Constructor & Destructor Documentation . . . 157
7.53.3 Member Function Documentation . . . 157
7.54 ElementManager Class Reference . . . 159
7.54.1 Detailed Description . . . 160
7.54.2 Constructor & Destructor Documentation . . . 160
7.54.3 Member Function Documentation . . . 160
7.54.4 Member Data Documentation . . . 163
7.55 ElementManager_if Class Reference . . . 164
7.55.1 Detailed Description . . . 164
7.55.2 Constructor & Destructor Documentation . . . 164
7.56 Enter Class Reference . . . 165
7.56.1 Detailed Description . . . 166
7.56.2 Constructor & Destructor Documentation . . . 167
7.56.3 Member Function Documentation . . . 167
7.56.4 Member Data Documentation . . . 169
7.57 Entity Class Reference . . . 170
7.57.1 Detailed Description . . . 171
7.57.2 Constructor & Destructor Documentation . . . 172
7.57.3 Member Function Documentation . . . 172
7.57.4 Friends And Related Function Documentation . . . 175
7.57.5 Member Data Documentation . . . 175
7.58 EntityGroup Class Reference . . . 176
7.58.1 Detailed Description . . . 177
7.58.2 Constructor & Destructor Documentation . . . 177
7.58.3 Member Function Documentation . . . 178
7.58.4 Member Data Documentation . . . 180
7.59 EntityType Class Reference . . . 180
7.59.1 Detailed Description . . . 182
7.59.2 Constructor & Destructor Documentation . . . 183
7.59.3 Member Function Documentation . . . 183
7.59.4 Member Data Documentation . . . 186
7.60 Event Class Reference . . . 187
7.60.1 Detailed Description . . . 187

7.60.2 Constructor & Destructor Documentation . . . 188
7.60.3 Member Function Documentation . . . 188
7.60.4 Member Data Documentation . . . 189
7.61 Exact Class Reference . . . 189
7.61.1 Detailed Description . . . 190
7.61.2 Constructor & Destructor Documentation . . . 190
7.61.3 Member Function Documentation . . . 191
7.61.4 Member Data Documentation . . . 193
7.62 Exit Class Reference . . . 194
7.62.1 Detailed Description . . . 195
7.62.2 Constructor & Destructor Documentation . . . 195
7.62.3 Member Function Documentation . . . 195
7.63 ExperimentManager Class Reference . . . 197
7.63.1 Detailed Description . . . 197
7.63.2 Constructor & Destructor Documentation . . . 198
7.63.3 Member Function Documentation . . . 198
7.63.4 Member Data Documentation . . . 199
7.64 ExperimentManager_if Class Reference . . . 200
7.64.1 Detailed Description . . . 202
7.64.2 Member Function Documentation . . . 202
7.65 ExperimentManagerDefaultImpl1 Class Reference . . . 203
7.65.1 Detailed Description . . . 205
7.65.2 Constructor & Destructor Documentation . . . 205
7.65.3 Member Function Documentation . . . 205
7.65.4 Member Data Documentation . . . 207
7.66 Failure Class Reference . . . 207
7.66.1 Detailed Description . . . 209
7.66.2 Constructor & Destructor Documentation . . . 209
7.66.3 Member Function Documentation . . . 210
7.67 File Class Reference . . . 211
7.67.1 Detailed Description . . . 213
7.67.2 Constructor & Destructor Documentation . . . 213
7.67.3 Member Function Documentation . . . 214
7.68 Ui::formPlugins Class Reference . . . 215
7.68.1 Detailed Description . . . 216
7.69 FullSimulationOfComplexModel Class Reference . . . 216
7.69.1 Detailed Description . . . 218
7.69.2 Constructor & Destructor Documentation . . . 218
7.69.3 Member Function Documentation . . . 218
7.70 ParserManager::GenerateNewParserResult Struct Reference . . . 218
7.70.1 Detailed Description . . . 219
7.70.2 Member Data Documentation . . . 219

7.71 GenesysApplication_if Class Reference . . . 220
7.71.1 Detailed Description . . . 220
7.71.2 Member Function Documentation . . . 220
7.72 genesyspp_driver Class Reference . . . 220
7.72.1 Detailed Description . . . 221
7.72.2 Constructor & Destructor Documentation . . . 221
7.72.3 Member Function Documentation . . . 222
7.72.4 Member Data Documentation . . . 225
7.73 yy::genesyspp_parser Class Reference . . . 226
7.73.1 Detailed Description . . . 230
7.73.2 Member Typedef Documentation . . . 230
7.73.3 Member Enumeration Documentation . . . 231
7.73.4 Constructor & Destructor Documentation . . . 232
7.73.5 Member Function Documentation . . . 232
7.73.6 Member Data Documentation . . . 252
7.74 GenesysQtGUI Class Reference . . . 255
7.74.1 Detailed Description . . . 256
7.74.2 Constructor & Destructor Documentation . . . 256
7.74.3 Member Function Documentation . . . 256
7.75 GenesysShell_if Class Reference . . . 257
7.75.1 Detailed Description . . . 258
7.75.2 Member Function Documentation . . . 258
7.76 GenesysTerminalApp Class Reference . . . 262
7.76.1 Detailed Description . . . 263
7.76.2 Constructor & Destructor Documentation . . . 263
7.76.3 Member Function Documentation . . . 263
7.76.4 Member Data Documentation . . . 264
7.77 Hold Class Reference . . . 265
7.77.1 Detailed Description . . . 266
7.77.2 Constructor & Destructor Documentation . . . 266
7.77.3 Member Function Documentation . . . 267
7.78 HypothesisTester_if Class Reference . . . 268
7.78.1 Detailed Description . . . 270
7.78.2 Member Enumeration Documentation . . . 270
7.78.3 Member Function Documentation . . . 271
7.79 HypothesisTesterDefaultImpl1 Class Reference . . . 275
7.79.1 Detailed Description . . . 277
7.79.2 Constructor & Destructor Documentation . . . 277
7.79.3 Member Function Documentation . . . 278
7.80 Leave Class Reference . . . 283
7.80.1 Detailed Description . . . 285
7.80.2 Constructor & Destructor Documentation . . . 286

7.80.3 Member Function Documentation . . . 286
7.80.4 Member Data Documentation . . . 288
7.81 LicenceManager Class Reference . . . 288
7.81.1 Detailed Description . . . 289
7.81.2 Constructor & Destructor Documentation . . . 289
7.81.3 Member Function Documentation . . . 289
7.81.4 Member Data Documentation . . . 291
7.82 List< T > Class Template Reference . . . 292
7.82.1 Detailed Description . . . 293
7.82.2 Member Typedef Documentation . . . 293
7.82.3 Constructor & Destructor Documentation . . . 293
7.82.4 Member Function Documentation . . . 293
7.82.5 Member Data Documentation . . . 296
7.83 yy::location Class Reference . . . 297
7.83.1 Detailed Description . . . 298
7.83.2 Member Typedef Documentation . . . 298
7.83.3 Constructor & Destructor Documentation . . . 299
7.83.4 Member Function Documentation . . . 299
7.83.5 Member Data Documentation . . . 300
7.84 LSODE Class Reference . . . 301
7.84.1 Detailed Description . . . 302
7.84.2 Constructor & Destructor Documentation . . . 302
7.84.3 Member Function Documentation . . . 303
7.84.4 Member Data Documentation . . . 305
7.85 MainWindow Class Reference . . . 306
7.85.1 Detailed Description . . . 308
7.85.2 Constructor & Destructor Documentation . . . 309
7.85.3 Member Function Documentation . . . 309
7.85.4 Member Data Documentation . . . 314
7.86 Ui::MainWindow Class Reference . . . 315
7.86.1 Detailed Description . . . 316
7.87 MapKernelAndUI Class Reference . . . 316
7.87.1 Detailed Description . . . 317
7.87.2 Constructor & Destructor Documentation . . . 317
7.87.3 Member Function Documentation . . . 317
7.87.4 Member Data Documentation . . . 318
7.88 MarkovChain Class Reference . . . 319
7.88.1 Detailed Description . . . 321
7.88.2 Constructor & Destructor Documentation . . . 321
7.88.3 Member Function Documentation . . . 322
7.88.4 Member Data Documentation . . . 324
7.89 Match Class Reference . . . 325

7.89.1 Detailed Description . . . 327
7.89.2 Constructor & Destructor Documentation . . . 327
7.89.3 Member Function Documentation . . . 327
7.90 Model Class Reference . . . 329
7.90.1 Detailed Description . . . 331
7.90.2 Constructor & Destructor Documentation . . . 331
7.90.3 Member Function Documentation . . . 331
7.90.4 Member Data Documentation . . . 336
7.91 ModelChecker_if Class Reference . . . 339
7.91.1 Detailed Description . . . 339
7.91.2 Member Function Documentation . . . 340
7.92 ModelCheckerDefaultImpl1 Class Reference . . . 341
7.92.1 Detailed Description . . . 342
7.92.2 Constructor & Destructor Documentation . . . 342
7.92.3 Member Function Documentation . . . 342
7.92.4 Member Data Documentation . . . 343
7.93 ModelComponent Class Reference . . . 344
7.93.1 Detailed Description . . . 345
7.93.2 Constructor & Destructor Documentation . . . 346
7.93.3 Member Function Documentation . . . 346
7.93.4 Member Data Documentation . . . 348
7.94 ModelElement Class Reference . . . 349
7.94.1 Detailed Description . . . 350
7.94.2 Constructor & Destructor Documentation . . . 350
7.94.3 Member Function Documentation . . . 351
7.94.4 Member Data Documentation . . . 355
7.95 ModelInfo Class Reference . . . 357
7.95.1 Detailed Description . . . 357
7.95.2 Constructor & Destructor Documentation . . . 357
7.95.3 Member Function Documentation . . . 358
7.95.4 Member Data Documentation . . . 359
7.96 ModelManager Class Reference . . . 360
7.96.1 Detailed Description . . . 361
7.96.2 Constructor & Destructor Documentation . . . 361
7.96.3 Member Function Documentation . . . 361
7.96.4 Member Data Documentation . . . 363
7.97 ModelPersistence_if Class Reference . . . 364
7.97.1 Detailed Description . . . 364
7.97.2 Member Function Documentation . . . 365
7.98 ModelPersistenceDefaultImpl1 Class Reference . . . 365
7.98.1 Detailed Description . . . 367
7.98.2 Constructor & Destructor Documentation . . . 367

7.98.3 Member Function Documentation . . . 367
7.98.4 Member Data Documentation . . . 369
7.99 ModelSimulation Class Reference . . . 370
7.99.1 Detailed Description . . . 373
7.99.2 Constructor & Destructor Documentation . . . 373
7.99.3 Member Function Documentation . . . 373
7.99.4 Friends And Related Function Documentation . . . 382
7.99.5 Member Data Documentation . . . 382
7.100 ParserManager::NewParser Struct Reference . . . 388
7.100.1 Detailed Description . . . 388
7.100.2 Member Data Documentation . . . 388
7.101 obj_t Class Reference . . . 389
7.101.1 Detailed Description . . . 389
7.101.2 Constructor & Destructor Documentation . . . 389
7.101.3 Member Data Documentation . . . 390
7.102 ODEfunction Class Reference . . . 391
7.102.1 Detailed Description . . . 391
7.102.2 Constructor & Destructor Documentation . . . 391
7.102.3 Member Data Documentation . . . 391
7.103 OLD_ODEelement Class Reference . . . 392
7.103.1 Detailed Description . . . 394
7.103.2 Constructor & Destructor Documentation . . . 394
7.103.3 Member Function Documentation . . . 394
7.103.4 Member Data Documentation . . . 396
7.104 OnEventManager Class Reference . . . 397
7.104.1 Detailed Description . . . 398
7.104.2 Constructor & Destructor Documentation . . . 399
7.104.3 Member Function Documentation . . . 399
7.104.4 Member Data Documentation . . . 405
7.105 OperatingSystem02 Class Reference . . . 409
7.105.1 Detailed Description . . . 410
7.105.2 Constructor & Destructor Documentation . . . 410
7.105.3 Member Function Documentation . . . 410
7.106 OperatingSystem03 Class Reference . . . 411
7.106.1 Detailed Description . . . 412
7.106.2 Constructor & Destructor Documentation . . . 412
7.106.3 Member Function Documentation . . . 412
7.107 Parser_if Class Reference . . . 413
7.107.1 Detailed Description . . . 413
7.107.2 Member Function Documentation . . . 413
7.108 ParserChangesInformation Class Reference . . . 414
7.108.1 Detailed Description . . . 415

7.108.2 Constructor & Destructor Documentation . . . 415
7.108.3 Member Data Documentation . . . 415
7.109 ParserDefaultImpl1 Class Reference . . . 417
7.109.1 Detailed Description . . . 417
7.109.2 Constructor & Destructor Documentation . . . 418
7.109.3 Member Function Documentation . . . 418
7.109.4 Member Data Documentation . . . 418
7.110 ParserDefaultImpl2 Class Reference . . . 419
7.110.1 Detailed Description . . . 420
7.110.2 Constructor & Destructor Documentation . . . 420
7.110.3 Member Function Documentation . . . 420
7.110.4 Member Data Documentation . . . 421
7.111 ParserManager Class Reference . . . 421
7.111.1 Detailed Description . . . 422
7.111.2 Constructor & Destructor Documentation . . . 422
7.111.3 Member Function Documentation . . . 422
7.112 PersistentObject_base Class Reference . . . 423
7.112.1 Detailed Description . . . 423
7.112.2 Constructor & Destructor Documentation . . . 423
7.112.3 Member Function Documentation . . . 423
7.113 PickStation Class Reference . . . 424
7.113.1 Detailed Description . . . 426
7.113.2 Constructor & Destructor Documentation . . . 426
7.113.3 Member Function Documentation . . . 427
7.114 PickUp Class Reference . . . 428
7.114.1 Detailed Description . . . 430
7.114.2 Constructor & Destructor Documentation . . . 430
7.114.3 Member Function Documentation . . . 430
7.115 Plugin Class Reference . . . 432
7.115.1 Detailed Description . . . 433
7.115.2 Constructor & Destructor Documentation . . . 433
7.115.3 Member Function Documentation . . . 433
7.115.4 Member Data Documentation . . . 434
7.116 PluginConnector_if Class Reference . . . 435
7.116.1 Detailed Description . . . 436
7.116.2 Member Function Documentation . . . 436
7.117 PluginConnectorDummyImpl1 Class Reference . . . 437
7.117.1 Detailed Description . . . 437
7.117.2 Constructor & Destructor Documentation . . . 437
7.117.3 Member Function Documentation . . . 438
7.118 PluginInformation Class Reference . . . 439
7.118.1 Detailed Description . . . 440

7.118.2 Constructor & Destructor Documentation . . . 440
7.118.3 Member Function Documentation . . . 441
7.118.4 Member Data Documentation . . . 445
7.119 PluginManager Class Reference . . . 448
7.119.1 Detailed Description . . . 449
7.119.2 Constructor & Destructor Documentation . . . 449
7.119.3 Member Function Documentation . . . 449
7.119.4 Member Data Documentation . . . 451
7.120 yy::position Class Reference . . . 451
7.120.1 Detailed Description . . . 452
7.120.2 Member Typedef Documentation . . . 452
7.120.3 Constructor & Destructor Documentation . . . 452
7.120.4 Member Function Documentation . . . 453
7.120.5 Member Data Documentation . . . 453
7.121 ProbabilityDistribution Class Reference . . . 454
7.121.1 Detailed Description . . . 455
7.121.2 Member Function Documentation . . . 455
7.121.3 Member Data Documentation . . . 460
7.122 Process Class Reference . . . 460
7.122.1 Detailed Description . . . 462
7.122.2 Constructor & Destructor Documentation . . . 462
7.122.3 Member Function Documentation . . . 463
7.122.4 Member Data Documentation . . . 466
7.123 QMainWindow Class Reference . . . 467
7.124 Queue Class Reference . . . 468
7.124.1 Detailed Description . . . 470
7.124.2 Member Enumeration Documentation . . . 470
7.124.3 Constructor & Destructor Documentation . . . 470
7.124.4 Member Function Documentation . . . 471
7.124.5 Member Data Documentation . . . 474
7.125 QueueableItem Class Reference . . . 475
7.125.1 Detailed Description . . . 476
7.125.2 Member Enumeration Documentation . . . 476
7.125.3 Constructor & Destructor Documentation . . . 476
7.125.4 Member Function Documentation . . . 476
7.125.5 Member Data Documentation . . . 478
7.126 Record Class Reference . . . 479
7.126.1 Detailed Description . . . 481
7.126.2 Constructor & Destructor Documentation . . . 482
7.126.3 Member Function Documentation . . . 482
7.126.4 Member Data Documentation . . . 484
7.127 Release Class Reference . . . 485

7.127.1 Detailed Description . . . 487
7.127.2 Constructor & Destructor Documentation . . . 488
7.127.3 Member Function Documentation . . . 488
7.127.4 Member Data Documentation . . . 490
7.128 Remove Class Reference . . . 490
7.128.1 Detailed Description . . . 492
7.128.2 Constructor & Destructor Documentation . . . 492
7.128.3 Member Function Documentation . . . 492
7.129 Resource Class Reference . . . 494
7.129.1 Detailed Description . . . 497
7.129.2 Member Typedef Documentation . . . 497
7.129.3 Member Enumeration Documentation . . . 498
7.129.4 Constructor & Destructor Documentation . . . 498
7.129.5 Member Function Documentation . . . 498
7.129.6 Member Data Documentation . . . 502
7.130 Sampler_if::RNG_Parameters Struct Reference . . . 504
7.130.1 Detailed Description . . . 505
7.130.2 Constructor & Destructor Documentation . . . 505
7.131 Route Class Reference . . . 505
7.131.1 Detailed Description . . . 508
7.131.2 Member Enumeration Documentation . . . 508
7.131.3 Constructor & Destructor Documentation . . . 508
7.131.4 Member Function Documentation . . . 509
7.131.5 Member Data Documentation . . . 511
7.132 Sampler_if Class Reference . . . 512
7.132.1 Detailed Description . . . 514
7.132.2 Member Function Documentation . . . 514
7.133 SamplerBoostImpl Class Reference . . . 517
7.133.1 Detailed Description . . . 519
7.133.2 Constructor & Destructor Documentation . . . 519
7.133.3 Member Function Documentation . . . 519
7.134 SamplerDefaultImpl1 Class Reference . . . 522
7.134.1 Detailed Description . . . 524
7.134.2 Constructor & Destructor Documentation . . . 524
7.134.3 Member Function Documentation . . . 524
7.134.4 Member Data Documentation . . . 528
7.135 ScenarioExperiment_if Class Reference . . . 529
7.135.1 Detailed Description . . . 529
7.136 Schedule Class Reference . . . 530
7.136.1 Detailed Description . . . 531
7.136.2 Constructor & Destructor Documentation . . . 531
7.136.3 Member Function Documentation . . . 532

7.137 Search Class Reference . . . 533
7.137.1 Detailed Description . . . 535
7.137.2 Constructor & Destructor Documentation . . . 535
7.137.3 Member Function Documentation . . . 536
7.138 SeizableItem Class Reference . . . 537
7.138.1 Detailed Description . . . 538
7.138.2 Member Enumeration Documentation . . . 538
7.138.3 Constructor & Destructor Documentation . . . 539
7.138.4 Member Function Documentation . . . 539
7.138.5 Member Data Documentation . . . 542
7.139 Seize Class Reference . . . 544
7.139.1 Detailed Description . . . 547
7.139.2 Constructor & Destructor Documentation . . . 547
7.139.3 Member Function Documentation . . . 548
7.139.4 Member Data Documentation . . . 551
7.140 yy::genesyspp_parser::semantic_type Class Reference . . . 552
7.140.1 Detailed Description . . . 553
7.140.2 Member Typedef Documentation . . . 554
7.140.3 Member Enumeration Documentation . . . 554
7.140.4 Constructor & Destructor Documentation . . . 554
7.140.5 Member Function Documentation . . . 555
7.140.6 Member Data Documentation . . . 557
7.141 Separate Class Reference . . . 558
7.141.1 Detailed Description . . . 560
7.141.2 Constructor & Destructor Documentation . . . 560
7.141.3 Member Function Documentation . . . 561
7.142 Sequence Class Reference . . . 562
7.142.1 Detailed Description . . . 564
7.142.2 Constructor & Destructor Documentation . . . 564
7.142.3 Member Function Documentation . . . 565
7.142.4 Member Data Documentation . . . 566
7.143 SequenceStep Class Reference . . . 566
7.143.1 Detailed Description . . . 567
7.143.2 Constructor & Destructor Documentation . . . 567
7.143.3 Member Function Documentation . . . 567
7.143.4 Member Data Documentation . . . 568
7.144 Set Class Reference . . . 569
7.144.1 Detailed Description . . . 571
7.144.2 Constructor & Destructor Documentation . . . 571
7.144.3 Member Function Documentation . . . 572
7.144.4 Member Data Documentation . . . 573
7.145 GenesysTerminalApp::ShellCommand Class Reference . . . 574

7.145.1 Detailed Description . . . 574
7.145.2 Constructor & Destructor Documentation . . . 574
7.145.3 Member Data Documentation . . . 574
7.146 Signal Class Reference . . . 575
7.146.1 Detailed Description . . . 577
7.146.2 Constructor & Destructor Documentation . . . 577
7.146.3 Member Function Documentation . . . 577
7.147 SimulationControl Class Reference . . . 579
7.147.1 Detailed Description . . . 580
7.147.2 Constructor & Destructor Documentation . . . 580
7.147.3 Member Function Documentation . . . 580
7.147.4 Member Data Documentation . . . 581
7.148 SimulationEvent Class Reference . . . 581
7.148.1 Detailed Description . . . 582
7.148.2 Constructor & Destructor Documentation . . . 582
7.148.3 Member Function Documentation . . . 582
7.148.4 Friends And Related Function Documentation . . . 586
7.148.5 Member Data Documentation . . . 586
7.149 SimulationExperiment Class Reference . . . 588
7.149.1 Detailed Description . . . 588
7.149.2 Constructor & Destructor Documentation . . . 588
7.150 SimulationReporter_if Class Reference . . . 589
7.150.1 Detailed Description . . . 589
7.150.2 Member Function Documentation . . . 589
7.151 SimulationReporterDefaultImpl1 Class Reference . . . 590
7.151.1 Detailed Description . . . 592
7.151.2 Constructor & Destructor Documentation . . . 592
7.151.3 Member Function Documentation . . . 592
7.151.4 Member Data Documentation . . . 593
7.152 SimulationResponse Class Reference . . . 594
7.152.1 Detailed Description . . . 595
7.152.2 Constructor & Destructor Documentation . . . 595
7.152.3 Member Function Documentation . . . 595
7.152.4 Member Data Documentation . . . 596
7.153 SimulationScenario Class Reference . . . 597
7.153.1 Detailed Description . . . 598
7.153.2 Constructor & Destructor Documentation . . . 598
7.153.3 Member Function Documentation . . . 598
7.153.4 Member Data Documentation . . . 600
7.154 Simulator Class Reference . . . 601
7.154.1 Detailed Description . . . 602
7.154.2 Member Typedef Documentation . . . 602

7.154.3 Constructor & Destructor Documentation . . . 602
7.154.4 Member Function Documentation . . . 602
7.154.5 Member Data Documentation . . . 604
7.155 SinkModelComponent Class Reference . . . 605
7.155.1 Detailed Description . . . 607
7.155.2 Constructor & Destructor Documentation . . . 607
7.155.3 Member Function Documentation . . . 607
7.156 yy::genesyspp_parser::stack< T, S >::slice Class Reference . . . 608
7.156.1 Detailed Description . . . 609
7.156.2 Constructor & Destructor Documentation . . . 609
7.156.3 Member Function Documentation . . . 609
7.156.4 Member Data Documentation . . . 609
7.157 Smart_AssignWriteSeizes Class Reference . . . 610
7.157.1 Detailed Description . . . 612
7.157.2 Constructor & Destructor Documentation . . . 612
7.157.3 Member Function Documentation . . . 612
7.158 Smart_Delay Class Reference . . . 613
7.158.1 Detailed Description . . . 614
7.158.2 Constructor & Destructor Documentation . . . 614
7.158.3 Member Function Documentation . . . 614
7.159 Smart_Dummy Class Reference . . . 615
7.159.1 Detailed Description . . . 616
7.159.2 Constructor & Destructor Documentation . . . 616
7.159.3 Member Function Documentation . . . 616
7.160 Smart_HoldSignal Class Reference . . . 617
7.160.1 Detailed Description . . . 618
7.160.2 Constructor & Destructor Documentation . . . 618
7.160.3 Member Function Documentation . . . 618
7.161 Smart_ModelInfoModelSimulation Class Reference . . . 619
7.161.1 Detailed Description . . . 620
7.161.2 Constructor & Destructor Documentation . . . 620
7.161.3 Member Function Documentation . . . 620
7.162 Smart_OnEvent Class Reference . . . 621
7.162.1 Detailed Description . . . 622
7.162.2 Constructor & Destructor Documentation . . . 622
7.162.3 Member Function Documentation . . . 622
7.163 Smart_Parser Class Reference . . . 625
7.163.1 Detailed Description . . . 626
7.163.2 Constructor & Destructor Documentation . . . 626
7.163.3 Member Function Documentation . . . 626
7.164 Smart_ParserModelFunctions Class Reference . . . 627
7.164.1 Detailed Description . . . 628

7.164.2 Constructor & Destructor Documentation . . . 628
7.164.3 Member Function Documentation . . . 628
7.165 Smart_Process Class Reference . . . 629
7.165.1 Detailed Description . . . 630
7.165.2 Constructor & Destructor Documentation . . . 630
7.165.3 Member Function Documentation . . . 630
7.166 Smart_ProcessSet Class Reference . . . 631
7.166.1 Detailed Description . . . 632
7.166.2 Constructor & Destructor Documentation . . . 632
7.166.3 Member Function Documentation . . . 632
7.167 Smart_RouteStation Class Reference . . . 633
7.167.1 Detailed Description . . . 634
7.167.2 Constructor & Destructor Documentation . . . 634
7.167.3 Member Function Documentation . . . 634
7.168 Smart_SeizeDelayRelease Class Reference . . . 635
7.168.1 Detailed Description . . . 636
7.168.2 Constructor & Destructor Documentation . . . 636
7.168.3 Member Function Documentation . . . 636
7.169 Smart_SeizeDelayReleaseMany Class Reference . . . 637
7.169.1 Detailed Description . . . 638
7.169.2 Constructor & Destructor Documentation . . . 638
7.169.3 Member Function Documentation . . . 638
7.170 Smart_Sequence Class Reference . . . 639
7.170.1 Detailed Description . . . 640
7.170.2 Constructor & Destructor Documentation . . . 640
7.170.3 Member Function Documentation . . . 640
7.171 Solver_if Class Reference . . . 641
7.171.1 Detailed Description . . . 642
7.171.2 Member Function Documentation . . . 642
7.172 SolverDefaultImpl1 Class Reference . . . 645
7.172.1 Detailed Description . . . 646
7.172.2 Constructor & Destructor Documentation . . . 646
7.172.3 Member Function Documentation . . . 646
7.172.4 Member Data Documentation . . . 649
7.173 SortFile Class Reference . . . 650
7.173.1 Detailed Description . . . 650
7.173.2 Constructor & Destructor Documentation . . . 650
7.173.3 Member Function Documentation . . . 651
7.173.4 Member Data Documentation . . . 652
7.174 SourceModelComponent Class Reference . . . 652
7.174.1 Detailed Description . . . 654
7.174.2 Constructor & Destructor Documentation . . . 654

7.174.3 Member Function Documentation . . . 655
7.174.4 Member Data Documentation . . . 658
7.175 yy::genesyspp_parser::stack< T, S > Class Template Reference . . . 659
7.175.1 Detailed Description . . . 660
7.175.2 Member Typedef Documentation . . . 660
7.175.3 Constructor & Destructor Documentation . . . 661
7.175.4 Member Function Documentation . . . 661
7.175.5 Member Data Documentation . . . 663
7.176 yy::genesyspp_parser::stack_symbol_type Struct Reference . . . 664
7.176.1 Detailed Description . . . 665
7.176.2 Member Typedef Documentation . . . 665
7.176.3 Constructor & Destructor Documentation . . . 665
7.176.4 Member Function Documentation . . . 666
7.177 Start Class Reference . . . 667
7.177.1 Detailed Description . . . 668
7.177.2 Constructor & Destructor Documentation . . . 668
7.177.3 Member Function Documentation . . . 668
7.178 Station Class Reference . . . 670
7.178.1 Detailed Description . . . 672
7.178.2 Constructor & Destructor Documentation . . . 672
7.178.3 Member Function Documentation . . . 673
7.178.4 Member Data Documentation . . . 674
7.179 Statistics_if Class Reference . . . 675
7.179.1 Detailed Description . . . 677
7.179.2 Member Function Documentation . . . 677
7.180 StatisticsCollector Class Reference . . . 679
7.180.1 Detailed Description . . . 681
7.180.2 Constructor & Destructor Documentation . . . 681
7.180.3 Member Function Documentation . . . 682
7.180.4 Member Data Documentation . . . 683
7.181 StatisticsDatafile_if Class Reference . . . 684
7.181.1 Detailed Description . . . 685
7.181.2 Member Function Documentation . . . 685
7.182 StatisticsDatafileDefaultImpl1 Class Reference . . . 687
7.182.1 Detailed Description . . . 689
7.182.2 Constructor & Destructor Documentation . . . 689
7.182.3 Member Function Documentation . . . 690
7.182.4 Member Data Documentation . . . 694
7.183 StatisticsDefaultImpl1 Class Reference . . . 701
7.183.1 Detailed Description . . . 703
7.183.2 Constructor & Destructor Documentation . . . 703
7.183.3 Member Function Documentation . . . 704

7.183.4 Member Data Documentation . . . 706
7.184 Stop Class Reference . . . 708
7.184.1 Detailed Description . . . 710
7.184.2 Constructor & Destructor Documentation . . . 710
7.184.3 Member Function Documentation . . . 710
7.185 Storage Class Reference . . . 712
7.185.1 Detailed Description . . . 714
7.185.2 Constructor & Destructor Documentation . . . 714
7.185.3 Member Function Documentation . . . 714
7.186 Store Class Reference . . . 716
7.186.1 Detailed Description . . . 717
7.186.2 Constructor & Destructor Documentation . . . 717
7.186.3 Member Function Documentation . . . 717
7.187 Submodel Class Reference . . . 719
7.187.1 Detailed Description . . . 721
7.187.2 Constructor & Destructor Documentation . . . 721
7.187.3 Member Function Documentation . . . 721
7.188 yy::genesyspp_parser::symbol_kind Struct Reference . . . 723
7.188.1 Detailed Description . . . 724
7.188.2 Member Enumeration Documentation . . . 724
7.189 yy::genesyspp_parser::symbol_type Struct Reference . . . 726
7.189.1 Detailed Description . . . 728
7.189.2 Member Typedef Documentation . . . 728
7.189.3 Constructor & Destructor Documentation . . . 728
7.190 yy::genesyspp_parser::syntax_error Struct Reference . . . 729
7.190.1 Detailed Description . . . 730
7.190.2 Constructor & Destructor Documentation . . . 730
7.190.3 Member Data Documentation . . . 730
7.191 HypothesisTester_if::TestResult Class Reference . . . 730
7.191.1 Detailed Description . . . 731
7.191.2 Constructor & Destructor Documentation . . . 731
7.191.3 Member Function Documentation . . . 731
7.191.4 Member Data Documentation . . . 732
7.192 yy::genesyspp_parser::token Struct Reference . . . 733
7.192.1 Detailed Description . . . 734
7.192.2 Member Typedef Documentation . . . 734
7.192.3 Member Enumeration Documentation . . . 734
7.193 TraceErrorEvent Class Reference . . . 736
7.193.1 Detailed Description . . . 737
7.193.2 Constructor & Destructor Documentation . . . 737
7.193.3 Member Function Documentation . . . 737
7.193.4 Member Data Documentation . . . 737

7.194 TraceEvent Class Reference . . . 738
7.194.1 Detailed Description . . . 738
7.194.2 Constructor & Destructor Documentation . . . 738
7.194.3 Member Function Documentation . . . 739
7.194.4 Member Data Documentation . . . 739
7.195 TraceManager Class Reference . . . 739
7.195.1 Detailed Description . . . 741
7.195.2 Constructor & Destructor Documentation . . . 741
7.195.3 Member Function Documentation . . . 741
7.195.4 Member Data Documentation . . . 744
7.196 TraceSimulationEvent Class Reference . . . 746
7.196.1 Detailed Description . . . 747
7.196.2 Constructor & Destructor Documentation . . . 747
7.196.3 Member Function Documentation . . . 748
7.196.4 Member Data Documentation . . . 748
7.197 TraceSimulationProcess Class Reference . . . 749
7.197.1 Detailed Description . . . 749
7.197.2 Constructor & Destructor Documentation . . . 750
7.198 Traits< T > Struct Template Reference . . . 750
7.198.1 Detailed Description . . . 750
7.199 Traits< GenesysApplication_if > Struct Reference . . . 750
7.199.1 Detailed Description . . . 751
7.199.2 Member Typedef Documentation . . . 751
7.199.3 Member Data Documentation . . . 751
7.200 Traits< HypothesisTester_if > Struct Reference . . . 751
7.200.1 Detailed Description . . . 751
7.200.2 Member Typedef Documentation . . . 752
7.200.3 Member Data Documentation . . . 752
7.201 Traits< Solver_if > Struct Reference . . . 752
7.201.1 Detailed Description . . . 752
7.201.2 Member Typedef Documentation . . . 752
7.201.3 Member Data Documentation . . . 753
7.202 TraitsKernel< T > Struct Template Reference . . . 753
7.202.1 Detailed Description . . . 753
7.202.2 Member Data Documentation . . . 753
7.203 TraitsKernel< Collector_if > Struct Reference . . . 754
7.203.1 Detailed Description . . . 754
7.203.2 Member Typedef Documentation . . . 754
7.204 TraitsKernel< Model > Struct Reference . . . 754
7.204.1 Detailed Description . . . 755
7.204.2 Member Data Documentation . . . 755
7.205 TraitsKernel< ModelChecker_if > Struct Reference . . . 755

7.205.1 Detailed Description . . . 755
7.205.2 Member Typedef Documentation . . . 755
7.205.3 Member Data Documentation . . . 756
7.206 TraitsKernel< ModelComponent > Struct Reference . . . 756
7.206.1 Detailed Description . . . 756
7.206.2 Member Typedef Documentation . . . 756
7.206.3 Member Data Documentation . . . 757
7.207 TraitsKernel< ModelElement > Struct Reference . . . 757
7.207.1 Detailed Description . . . 757
7.207.2 Member Data Documentation . . . 757
7.208 TraitsKernel< ModelPersistence_if > Struct Reference . . . 758
7.208.1 Detailed Description . . . 758
7.208.2 Member Typedef Documentation . . . 758
7.208.3 Member Data Documentation . . . 758
7.209 TraitsKernel< Parser_if > Struct Reference . . . 759
7.209.1 Detailed Description . . . 759
7.209.2 Member Typedef Documentation . . . 759
7.210 TraitsKernel< PluginConnector_if > Struct Reference . . . 759
7.210.1 Detailed Description . . . 760
7.210.2 Member Typedef Documentation . . . 760
7.210.3 Member Data Documentation . . . 760
7.211 TraitsKernel< Sampler_if > Struct Reference . . . 760
7.211.1 Detailed Description . . . 760
7.211.2 Member Typedef Documentation . . . 761
7.212 TraitsKernel< SimulationReporter_if > Struct Reference . . . 761
7.212.1 Detailed Description . . . 761
7.212.2 Member Typedef Documentation . . . 761
7.212.3 Member Data Documentation . . . 762
7.213 TraitsKernel< Statistics_if > Struct Reference . . . 762
7.213.1 Detailed Description . . . 762
7.213.2 Member Typedef Documentation . . . 762
7.213.3 Member Data Documentation . . . 763
7.214 TraitsKernel< StatisticsDatafile_if > Struct Reference . . . 763
7.214.1 Detailed Description . . . 763
7.214.2 Member Typedef Documentation . . . 763
7.214.3 Member Data Documentation . . . 764
7.215 Ui_DialogAbout Class Reference . . . 764
7.215.1 Detailed Description . . . 765
7.215.2 Member Function Documentation . . . 766
7.215.3 Member Data Documentation . . . 766
7.216 Ui_DialogModelInformation Class Reference . . . 767
7.216.1 Detailed Description . . . 769

7.216.2 Member Function Documentation . . . 769
7.216.3 Member Data Documentation . . . 769
7.217 Ui_formPlugins Class Reference . . . 771
7.217.1 Detailed Description . . . 772
7.217.2 Member Function Documentation . . . 772
7.217.3 Member Data Documentation . . . 772
7.218 Ui_MainWindow Class Reference . . . 773
7.218.1 Detailed Description . . . 775
7.218.2 Member Function Documentation . . . 775
7.218.3 Member Data Documentation . . . 776
7.219 yy::genesyspp_parser::semantic_type::union_type Union Reference . . . 789
7.219.1 Detailed Description . . . 789
7.219.2 Member Data Documentation . . . 790
7.220 Unstore Class Reference . . . 790
7.220.1 Detailed Description . . . 792
7.220.2 Constructor & Destructor Documentation . . . 792
7.220.3 Member Function Documentation . . . 792
7.221 Util Class Reference . . . 794
7.221.1 Detailed Description . . . 795
7.221.2 Member Typedef Documentation . . . 795
7.221.3 Member Enumeration Documentation . . . 795
7.221.4 Constructor & Destructor Documentation . . . 796
7.221.5 Member Function Documentation . . . 796
7.221.6 Member Data Documentation . . . 799
7.222 Variable Class Reference . . . 800
7.222.1 Detailed Description . . . 801
7.222.2 Constructor & Destructor Documentation . . . 802
7.222.3 Member Function Documentation . . . 802
7.222.4 Member Data Documentation . . . 805
7.223 Waiting Class Reference . . . 806
7.223.1 Detailed Description . . . 807
7.223.2 Constructor & Destructor Documentation . . . 807
7.223.3 Member Function Documentation . . . 807
7.223.4 Member Data Documentation . . . 808
7.224 WaitingResource Class Reference . . . 809
7.224.1 Detailed Description . . . 810
7.224.2 Constructor & Destructor Documentation . . . 810
7.224.3 Member Function Documentation . . . 810
7.224.4 Member Data Documentation . . . 810
7.225 Write Class Reference . . . 811
7.225.1 Detailed Description . . . 813
7.225.2 Member Enumeration Documentation . . . 813

7.225.3 Constructor & Destructor Documentation . . . 814
7.225.4 Member Function Documentation . . . 814
7.225.5 Member Data Documentation . . . 816
7.226 WriteExpression Class Reference . . . 817
7.226.1 Detailed Description . . . 817
7.226.2 Constructor & Destructor Documentation . . . 818
7.226.3 Member Data Documentation . . . 818
7.227 WritelnExpression Class Reference . . . 818
7.227.1 Detailed Description . . . 819
7.227.2 Constructor & Destructor Documentation . . . 819
7.227.3 Member Data Documentation . . . 819
7.228 WritelnText Class Reference . . . 820
7.228.1 Detailed Description . . . 820
7.228.2 Constructor & Destructor Documentation . . . 820
7.228.3 Member Data Documentation . . . 821
7.229 WriteText Class Reference . . . 821
7.229.1 Detailed Description . . . 822
7.229.2 Constructor & Destructor Documentation . . . 822
7.229.3 Member Data Documentation . . . 822
7.230 yy_buffer_state Struct Reference . . . 823
7.230.1 Detailed Description . . . 823
7.230.2 Member Data Documentation . . . 823
7.231 yy_trans_info Struct Reference . . . 825
7.231.1 Detailed Description . . . 825
7.231.2 Member Data Documentation . . . 825

8 File Documentation **826**

8.1 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩S/2019_2022_GenESyS/.dep.inc File Reference . . . 826
8.2 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩S/2019_2022_GenESyS/.dep.inc . . . 826
8.3 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩S/2019_2022_GenESyS/applications/BaseConsoleGenesysApplication.cpp File Reference . . . 826
8.4 BaseConsoleGenesysApplication.cpp . . . 826
8.5 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩S/2019_2022_GenESyS/applications/BaseConsoleGenesysApplication.h File Reference . . . 829
8.6 BaseConsoleGenesysApplication.h . . . 829
8.7 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩S/2019_2022_GenESyS/applications/GenesysApplication_if.h File Reference . . . 830
8.8 GenesysApplication_if.h . . . 830
8.9 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩S/2019_2022_GenESyS/applications/gui/qt/GenesysQtGUI.cpp File Reference . . . 831
8.10 GenesysQtGUI.cpp . . . 831

8.11 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/gui/qt/GenesysQtGUI.h File Reference 831
8.12 GenesysQtGUI.h 832
8.13 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.cpp File Reference 832
8.14 mainwindow.cpp 833
8.15 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.h File Reference 838
8.16 mainwindow.h 839
8.17 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/gui/qt/mainWindow/ui_mainwindow.h File Reference 840
8.18 ui_mainwindow.h 841
8.19 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/ui_mainwindow.h File Reference 850
8.20 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/ui_mainwindow.h 850
8.21 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/gui/qt/mapkernelandui.cpp File Reference 860
8.22 mapkernelandui.cpp 860
8.23 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/gui/qt/mapkernelandui.h File Reference 861
8.24 mapkernelandui.h 862
8.25 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/terminal/GenesysShell_if.h File Reference 862
8.26 GenesysShell_if.h 862
8.27 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/terminal/GenesysTerminalApp.cpp File Reference 863
8.28 GenesysTerminalApp.cpp 864
8.29 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/applications/terminal/GenesysTerminalApp.h File Reference 865
8.30 GenesysTerminalApp.h 865
8.31 html/dynsections.js File Reference 866
8.32 dynsections.js 866
8.33 html/jquery.js File Reference 868
8.34 jquery.js 868
8.35 html/menu.js File Reference 883
8.36 menu.js 883
8.37 html/menudata.js File Reference 883
8.38 menudata.js 883
8.39 html/search/search.js File Reference 884
8.40 search.js 884
8.41 html/search/searchdata.js File Reference 893
8.42 searchdata.js 893
8.43 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/book/Book_Cap02_Example01.cpp File Reference 894

8.44 Book_Cap02_Example01.cpp 894
8.45 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/book/Book_Cap02_Example01.h File Reference 895
8.46 Book_Cap02_Example01.h 895
8.47 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_AssignWriteSeizes.cpp File Reference 896
8.48 Smart_AssignWriteSeizes.cpp 896
8.49 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_AssignWriteSeizes.h File Reference 898
8.50 Smart_AssignWriteSeizes.h 899
8.51 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_Delay.cpp File Reference 899
8.52 Smart_Delay.cpp 899
8.53 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_Delay.h File Reference 900
8.54 Smart_Delay.h 901
8.55 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_Dummy.cpp File Reference 901
8.56 Smart_Dummy.cpp 902
8.57 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_Dummy.h File Reference 903
8.58 Smart_Dummy.h 903
8.59 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_HoldSignal.cpp File Reference 904
8.60 Smart_HoldSignal.cpp 904
8.61 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_HoldSignal.h File Reference 905
8.62 Smart_HoldSignal.h 906
8.63 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_ModelInfoModelSimulation.cpp File Reference . 906
8.64 Smart_ModelInfoModelSimulation.cpp 906
8.65 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_ModelInfoModelSimulation.h File Reference . . . 907
8.66 Smart_ModelInfoModelSimulation.h 908
8.67 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_OnEvent.cpp File Reference 909
8.68 Smart_OnEvent.cpp 909
8.69 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_OnEvent.h File Reference 910
8.70 Smart_OnEvent.h 911
8.71 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_Parser.cpp File Reference 912
8.72 Smart_Parser.cpp 912
8.73 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/smarts/Smart_Parser.h File Reference 913
8.74 Smart_Parser.h 913

8.75 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.cpp File Reference . . . 914
8.76 Smart_ParserModelFunctions.cpp . . . 914
8.77 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.h File Reference . . . 915
8.78 Smart_ParserModelFunctions.h . . . 916
8.79 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Process.cpp File Reference . . . 917
8.80 Smart_Process.cpp . . . 917
8.81 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Process.h File Reference . . . 918
8.82 Smart_Process.h . . . 918
8.83 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ProcessSet.cpp File Reference . . . 919
8.84 Smart_ProcessSet.cpp . . . 919
8.85 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ProcessSet.h File Reference . . . 920
8.86 Smart_ProcessSet.h . . . 921
8.87 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_RouteStation.cpp File Reference . . . 921
8.88 Smart_RouteStation.cpp . . . 921
8.89 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_RouteStation.h File Reference . . . 923
8.90 Smart_RouteStation.h . . . 923
8.91 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayRelease.cpp File Reference . . . 924
8.92 Smart_SeizeDelayRelease.cpp . . . 924
8.93 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayRelease.h File Reference . . . 925
8.94 Smart_SeizeDelayRelease.h . . . 926
8.95 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.cpp File Reference . . . 926
8.96 Smart_SeizeDelayReleaseMany.cpp . . . 927
8.97 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.h File Reference . . . 928
8.98 Smart_SeizeDelayReleaseMany.h . . . 929
8.99 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Sequence.cpp File Reference . . . 929
8.100 Smart_Sequence.cpp . . . 929
8.101 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Sequence.h File Reference . . . 931
8.102 Smart_Sequence.h . . . 931
8.103 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/teaching/AnElectronicAssemblyAndTestSystem.cpp File Reference . . . 932
8.104 AnElectronicAssemblyAndTestSystem.cpp . . . 932

8.105 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/AnElectronicAssemblyAndTestSystem.h File Reference 934

8.106 AnElectronicAssemblyAndTestSystem.h . . . 935

8.107 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/FullSimulationOfComplexModel.cpp File Reference . . 936

8.108 FullSimulationOfComplexModel.cpp . . . 936

8.109 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/FullSimulationOfComplexModel.h File Reference . . . 938

8.110 FullSimulationOfComplexModel.h . . . 938

8.111 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/OperatingSystem02.cpp File Reference . . . 939

8.112 OperatingSystem02.cpp . . . 939

8.113 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/OperatingSystem02.h File Reference . . . 941

8.114 OperatingSystem02.h . . . 942

8.115 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/OperatingSystem03.cpp File Reference . . . 942

8.116 OperatingSystem03.cpp . . . 943

8.117 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/examples/teaching/OperatingSystem03.h File Reference . . . 946

8.118 OperatingSystem03.h . . . 947

8.119 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Attribute.cpp File Reference . . . 948

8.120 Attribute.cpp . . . 948

8.121 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Attribute.h File Reference . . . 949

8.122 Attribute.h . . . 949

8.123 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ComponentManager.cpp File Reference . . . 950

8.124 ComponentManager.cpp . . . 950

8.125 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ComponentManager.h File Reference . . . 952

8.126 ComponentManager.h . . . 952

8.127 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ConnectionManager.cpp File Reference . . . 953

8.128 ConnectionManager.cpp . . . 953

8.129 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ConnectionManager.h File Reference . . . 954

8.129.1 Typedef Documentation . . . 955

8.130 ConnectionManager.h . . . 955

8.131 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Counter.cpp File Reference . . . 956

8.132 Counter.cpp . . . 956

8.133 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Counter.h File Reference . . . 958

8.134 Counter.h . . . 958

8.135 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/DefineGetterSetter.h File Reference 959
8.135.1 Typedef Documentation 960
8.135.2 Function Documentation 961
8.136 DefineGetterSetter.h 963
8.137 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ElementManager.cpp File Reference 965
8.138 ElementManager.cpp 965
8.139 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ElementManager.h File Reference 968
8.140 ElementManager.h 968
8.141 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ElementManager_if.h File Reference 969
8.142 ElementManager_if.h 969
8.143 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Entity.cpp File Reference 969
8.144 Entity.cpp 970
8.145 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Entity.h File Reference 972
8.146 Entity.h 973
8.147 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/EntityType.cpp File Reference 973
8.148 EntityType.cpp 974
8.149 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/EntityType.h File Reference 976
8.150 EntityType.h 976
8.151 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Event.cpp File Reference 977
8.152 Event.cpp 978
8.153 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Event.h File Reference 979
8.154 Event.h 979
8.155 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ExperimentManager.cpp File Reference 980
8.156 ExperimentManager.cpp 980
8.157 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ExperimentManager.h File Reference 981
8.158 ExperimentManager.h 982
8.159 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.cpp File Reference . . 983
8.160 ExperimentManagerDefaultImpl1.cpp 983
8.161 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.h File Reference 984
8.162 ExperimentManagerDefaultImpl1.h 985
8.163 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ExperimetManager_if.h File Reference 986

8.164 ExperimetManager_if.h 986
8.165 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/LicenceManager.cpp File Reference 987
8.166 LicenceManager.cpp 987
8.167 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/LicenceManager.h File Reference 988
8.168 LicenceManager.h 989
8.169 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Model.cpp File Reference 990
8.169.1 Function Documentation 990
8.170 Model.cpp 990
8.171 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Model.h File Reference 995
8.172 Model.h 996
8.173 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelChecker_if.h File Reference 997
8.174 ModelChecker_if.h 998
8.175 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.cpp File Reference 998
8.176 ModelCheckerDefaultImpl1.cpp 999
8.177 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.h File Reference 1002
8.178 ModelCheckerDefaultImpl1.h 1002
8.179 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelComponent.cpp File Reference 1003
8.180 ModelComponent.cpp 1003
8.181 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelComponent.h File Reference 1005
8.182 ModelComponent.h 1006
8.183 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelElement.cpp File Reference 1007
8.184 ModelElement.cpp 1007
8.185 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelElement.h File Reference 1011
8.186 ModelElement.h 1011
8.187 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelInfo.cpp File Reference 1012
8.188 ModelInfo.cpp 1013
8.189 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelInfo.h File Reference 1014
8.190 ModelInfo.h 1014
8.191 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelManager.cpp File Reference 1015
8.192 ModelManager.cpp 1015
8.193 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelManager.h File Reference 1017

8.194 ModelManager.h . . . 1017
8.195 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelPersistence_if.h File Reference . . . 1018
8.196 ModelPersistence_if.h . . . 1018
8.197 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelPersistenceDefaultImpl1.cpp File Reference . . . 1019
8.198 ModelPersistenceDefaultImpl1.cpp . . . 1019
8.199 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelPersistenceDefaultImpl1.h File Reference . . . 1025
8.200 ModelPersistenceDefaultImpl1.h . . . 1025
8.201 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelSimulation.cpp File Reference . . . 1026
8.202 ModelSimulation.cpp . . . 1026
8.203 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ModelSimulation.h File Reference . . . 1035
8.204 ModelSimulation.h . . . 1036
8.205 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/OnEventManager.cpp File Reference . . . 1038
8.206 OnEventManager.cpp . . . 1039
8.207 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/OnEventManager.h File Reference . . . 1041
8.207.1 Typedef Documentation . . . 1041
8.208 OnEventManager.h . . . 1042
8.209 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/Parser_if.h File Reference . . . 1045
8.210 Parser_if.h . . . 1046
8.211 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-driver.cpp File Reference . 1046
8.212 Genesys++-driver.cpp . . . 1047
8.213 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-driver.h File Reference . . . 1048
8.213.1 Macro Definition Documentation . . . 1049
8.213.2 Variable Documentation . . . 1049
8.214 Genesys++-driver.h . . . 1049
8.215 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-scanner.cpp File Reference 1050
8.215.1 Macro Definition Documentation . . . 1054
8.215.2 Typedef Documentation . . . 1064
8.215.3 Function Documentation . . . 1066
8.215.4 Variable Documentation . . . 1073
8.216 Genesys++-scanner.cpp . . . 1077
8.217 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.cpp File Reference . . . 1154
8.217.1 Macro Definition Documentation . . . 1155
8.218 GenesysParser.cpp . . . 1158

8.219 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h File Reference 1191
8.219.1 Detailed Description 1192
8.219.2 Macro Definition Documentation 1193
8.220 GenesysParser.h 1196
8.221 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/location.hh File Reference 1230
8.221.1 Detailed Description 1232
8.221.2 Macro Definition Documentation 1232
8.222 location.hh 1232
8.223 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.cpp File Reference 1235
8.224 obj_t.cpp 1236
8.225 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.h File Reference 1236
8.226 obj_t.h 1236
8.227 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/parser/bisonparser.yy File Reference . . 1237
8.228 bisonparser.yy 1237
8.229 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/parser/lexerparser.ll File Reference . . . 1244
8.230 lexerparser.ll 1244
8.231 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/position.hh File Reference 1249
8.232 position.hh 1249
8.233 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/stack.hh File Reference 1249
8.234 stack.hh 1249
8.235 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ParserChangesInformation.cpp File Reference 1250
8.236 ParserChangesInformation.cpp 1250
8.237 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ParserChangesInformation.h File Reference 1250
8.238 ParserChangesInformation.h 1251
8.239 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl1.cpp File Reference 1252
8.240 ParserDefaultImpl1.cpp 1252
8.241 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl1.h File Reference 1253
8.242 ParserDefaultImpl1.h 1253
8.243 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl2.cpp File Reference 1254
8.244 ParserDefaultImpl2.cpp 1254
8.245 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl2.h File Reference 1255
8.246 ParserDefaultImpl2.h 1256

8.247 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.cpp File Reference . . . 1257
8.248 ParserManager.cpp . . . 1257
8.249 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.h File Reference . . . 1257
8.250 ParserManager.h . . . 1258
8.251 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PersistentObject_base.h File Reference . . . 1259
8.252 PersistentObject_base.h . . . 1259
8.253 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Plugin.cpp File Reference . . . 1260
8.254 Plugin.cpp . . . 1260
8.255 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Plugin.h File Reference . . . 1261
8.256 Plugin.h . . . 1262
8.257 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginConnector_if.h File Reference . . . 1262
8.258 PluginConnector_if.h . . . 1263
8.259 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.cpp File Reference . . . 1263
8.260 PluginConnectorDummyImpl1.cpp . . . 1264
8.261 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.h File Reference . . . 1266
8.262 PluginConnectorDummyImpl1.h . . . 1267
8.263 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginInformation.cpp File Reference . . . 1267
8.264 PluginInformation.cpp . . . 1268
8.265 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginInformation.h File Reference . . . 1270
8.265.1 Typedef Documentation . . . 1271
8.266 PluginInformation.h . . . 1271
8.267 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginManager.cpp File Reference . . . 1273
8.268 PluginManager.cpp . . . 1273
8.269 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginManager.h File Reference . . . 1275
8.270 PluginManager.h . . . 1276
8.271 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ScenarioExperiment_if.h File Reference . . . 1276
8.272 ScenarioExperiment_if.h . . . 1276
8.273 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationControl.cpp File Reference . . . 1277
8.274 SimulationControl.cpp . . . 1277
8.275 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationControl.h File Reference . . . 1277
8.276 SimulationControl.h . . . 1278

8.277 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationExperiment.cpp File Reference 1278
8.278 SimulationExperiment.cpp 1279
8.279 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationExperiment.h File Reference 1279
8.280 SimulationExperiment.h 1279
8.281 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationReporter_if.h File Reference 1280
8.282 SimulationReporter_if.h 1280
8.283 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.cpp File Reference . . . 1281
8.284 SimulationReporterDefaultImpl1.cpp 1281
8.285 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.h File Reference 1285
8.286 SimulationReporterDefaultImpl1.h 1285
8.287 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationResponse.cpp File Reference 1286
8.288 SimulationResponse.cpp 1286
8.289 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationResponse.h File Reference 1287
8.290 SimulationResponse.h 1287
8.291 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationScenario.cpp File Reference 1288
8.292 SimulationScenario.cpp 1288
8.293 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationScenario.h File Reference 1290
8.294 SimulationScenario.h 1290
8.295 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Simulator.cpp File Reference 1291
8.295.1 Function Documentation 1291
8.296 Simulator.cpp 1292
8.297 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Simulator.h File Reference 1293
8.297.1 Typedef Documentation 1294
8.298 Simulator.h 1294
8.299 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SinkModelComponent.cpp File Reference 1295
8.300 SinkModelComponent.cpp 1295
8.301 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SinkModelComponent.h File Reference 1296
8.302 SinkModelComponent.h 1296
8.303 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SourceModelComponent.cpp File Reference 1297
8.304 SourceModelComponent.cpp 1297
8.305 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SourceModelComponent.h File Reference 1299

8.306 SourceModelComponent.h . . . 1300

8.307 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/StatisticsCollector.cpp File Reference . . . 1301

8.307.1 Typedef Documentation . . . 1301

8.308 StatisticsCollector.cpp . . . 1301

8.309 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/StatisticsCollector.h File Reference . . . 1303

8.310 StatisticsCollector.h . . . 1303

8.311 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/TraceManager.cpp File Reference . . . 1304

8.312 TraceManager.cpp . . . 1304

8.313 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/TraceManager.h File Reference . . . 1306

8.313.1 Typedef Documentation . . . 1306

8.314 TraceManager.h . . . 1308

8.315 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Collector_if.h File Reference . . . 1310

8.315.1 Typedef Documentation . . . 1311

8.315.2 Function Documentation . . . 1311

8.316 Collector_if.h . . . 1311

8.317 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafile_if.h File Reference . . . 1312

8.318 CollectorDatafile_if.h . . . 1312

8.319 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.cpp File Reference . . . 1313

8.320 CollectorDatafileDefaultImpl1.cpp . . . 1313

8.321 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.h File Reference . . . 1315

8.322 CollectorDatafileDefaultImpl1.h . . . 1315

8.323 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.cpp File Reference . . . 1316

8.324 CollectorDefaultImpl1.cpp . . . 1317

8.325 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.h File Reference . . . 1318

8.326 CollectorDefaultImpl1.h . . . 1318

8.327 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Sampler_if.h File Reference . . . 1319

8.328 Sampler_if.h . . . 1319

8.329 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerBoostImpl.cpp File Reference . . . 1320

8.330 SamplerBoostImpl.cpp . . . 1320

8.331 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerBoostImpl.h File Reference . . . 1321

8.332 SamplerBoostImpl.h . . . 1322

8.333 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.cpp File Reference . . . 1322

8.334 SamplerDefaultImpl1.cpp . . . 1323
8.335 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.h File Reference . . . 1326
8.336 SamplerDefaultImpl1.h . . . 1327
8.337 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/SorttFile.cpp File Reference . . . 1327
8.338 SorttFile.cpp . . . 1328
8.339 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/SorttFile.h File Reference . . . 1329
8.340 SorttFile.h . . . 1330
8.341 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/Statistics_if.h File Reference . . . 1330
8.342 Statistics_if.h . . . 1331
8.343 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/StatisticsDataFile_if.h File Reference . . . 1331
8.344 StatisticsDataFile_if.h . . . 1332
8.345 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.cpp File Reference . . . 1333
8.346 StatisticsDataFileDefaultImpl.cpp . . . 1333
8.347 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.h File Reference . . . 1337
8.347.1 Typedef Documentation . . . 1338
8.348 StatisticsDataFileDefaultImpl.h . . . 1338
8.349 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/StatisticsDefaultImpl1.cpp File Reference . . . 1339
8.350 StatisticsDefaultImpl1.cpp . . . 1340
8.351 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/statistics/StatisticsDefaultImpl1.h File Reference . . . 1341
8.352 StatisticsDefaultImpl1.h . . . 1342
8.353 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/TraitsKernel.h File Reference . . . 1343
8.354 TraitsKernel.h . . . 1344
8.355 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/util/Exact.h File Reference . . . 1345
8.356 Exact.h . . . 1346
8.357 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/util/List.h File Reference . . . 1347
8.358 List.h . . . 1348
8.359 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/util/Util.cpp File Reference . . . 1350
8.360 Util.cpp . . . 1351
8.361 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/kernel/util/Util.h File Reference . . . 1353
8.361.1 Function Documentation . . . 1354
8.362 Util.h . . . 1357

8.363 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/main.cpp File Reference . . . 1360
8.363.1 Function Documentation . . . 1361
8.364 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/main.cpp . . . 1361
8.365 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/nbproject/private/c_standard_headers_indexer.c File Reference . . . 1361
8.366 c_standard_headers_indexer.c . . . 1362
8.367 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/nbproject/private/cpp_standard_headers_indexer.cpp File Reference . . . 1363
8.368 cpp_standard_headers_indexer.cpp . . . 1364
8.369 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Access.cpp File Reference . . . 1366
8.370 Access.cpp . . . 1366
8.371 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Access.h File Reference . . . 1367
8.372 Access.h . . . 1367
8.373 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Assign.cpp File Reference . . . 1368
8.374 Assign.cpp . . . 1368
8.375 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Assign.h File Reference . . . 1370
8.376 Assign.h . . . 1370
8.377 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Batch.cpp File Reference . . . 1371
8.378 Batch.cpp . . . 1372
8.379 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Batch.h File Reference . . . 1373
8.380 Batch.h . . . 1373
8.381 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/CellularAutomata.cpp File Reference . . . 1374
8.382 CellularAutomata.cpp . . . 1374
8.383 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/CellularAutomata.h File Reference . . . 1375
8.384 CellularAutomata.h . . . 1376
8.385 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Create.cpp File Reference . . . 1376
8.386 Create.cpp . . . 1376
8.387 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Create.h File Reference . . . 1378
8.388 Create.h . . . 1379
8.389 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Decide.cpp File Reference . . . 1380
8.390 Decide.cpp . . . 1380
8.391 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Decide.h File Reference . . . 1382

8.392 Decide.h . . . 1382
8.393 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Delay.cpp File Reference . . . 1383
8.394 Delay.cpp . . . 1383
8.395 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Delay.h File Reference . . . 1385
8.396 Delay.h . . . 1386
8.397 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Dispose.cpp File Reference . . . 1386
8.398 Dispose.cpp . . . 1387
8.399 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Dispose.h File Reference . . . 1388
8.400 Dispose.h . . . 1389
8.401 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/DropOff.cpp File Reference . . . 1389
8.402 DropOff.cpp . . . 1390
8.403 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/DropOff.h File Reference . . . 1390
8.404 DropOff.h . . . 1391
8.405 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Dummy.cpp File Reference . . . 1391
8.406 Dummy.cpp . . . 1392
8.407 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Dummy.h File Reference . . . 1393
8.408 Dummy.h . . . 1393
8.409 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Enter.cpp File Reference . . . 1394
8.410 Enter.cpp . . . 1394
8.411 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Enter.h File Reference . . . 1395
8.412 Enter.h . . . 1396
8.413 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Exit.cpp File Reference . . . 1397
8.414 Exit.cpp . . . 1397
8.415 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Exit.h File Reference . . . 1398
8.416 Exit.h . . . 1398
8.417 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Hold.cpp File Reference . . . 1399
8.418 Hold.cpp . . . 1399
8.419 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Hold.h File Reference . . . 1400
8.420 Hold.h . . . 1401
8.421 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Leave.cpp File Reference . . . 1401
8.422 Leave.cpp . . . 1401

8.423 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Leave.h File Reference . . . 1403
8.424 Leave.h . . . 1403
8.425 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/LSODE.cpp File Reference . . . 1404
8.426 LSODE.cpp . . . 1404
8.427 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/LSODE.h File Reference . . . 1406
8.428 LSODE.h . . . 1407
8.429 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/MarkovChain.cpp File Reference . . . 1408
8.430 MarkovChain.cpp . . . 1408
8.431 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/MarkovChain.h File Reference . . . 1410
8.432 MarkovChain.h . . . 1411
8.433 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Match.cpp File Reference . . . 1411
8.434 Match.cpp . . . 1412
8.435 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Match.h File Reference . . . 1413
8.436 Match.h . . . 1413
8.437 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/OLD_ODEelement.cpp File Reference . . . 1414
8.438 OLD_ODEelement.cpp . . . 1414
8.439 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/OLD_ODEelement.h File Reference . . . 1415
8.440 OLD_ODEelement.h . . . 1416
8.441 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickStation.cpp File Reference . . . 1417
8.442 PickStation.cpp . . . 1417
8.443 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickStation.h File Reference . . . 1418
8.444 PickStation.h . . . 1419
8.445 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickUp.cpp File Reference . . . 1419
8.446 PickUp.cpp . . . 1419
8.447 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickUp.h File Reference . . . 1420
8.448 PickUp.h . . . 1421
8.449 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Process.cpp File Reference . . . 1421
8.450 Process.cpp . . . 1422
8.451 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Process.h File Reference . . . 1424
8.452 Process.h . . . 1424

8.453 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/QueueableItem.cpp File Reference 1425
8.454 QueueableItem.cpp 1426
8.455 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/QueueableItem.h File Reference 1428
8.456 QueueableItem.h 1428
8.457 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Record.cpp File Reference 1429
8.458 Record.cpp 1430
8.459 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Record.h File Reference 1432
8.460 Record.h 1432
8.461 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Release.cpp File Reference 1433
8.462 Release.cpp 1433
8.463 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Release.h File Reference 1435
8.464 Release.h 1436
8.465 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Remove.cpp File Reference 1437
8.466 Remove.cpp 1437
8.467 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Remove.h File Reference 1438
8.468 Remove.h 1439
8.469 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Route.cpp File Reference 1439
8.470 Route.cpp 1440
8.471 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Route.h File Reference 1442
8.472 Route.h 1443
8.473 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Search.cpp File Reference 1444
8.474 Search.cpp 1444
8.475 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Search.h File Reference 1445
8.476 Search.h 1445
8.477 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/SeizableItem.cpp File Reference 1446
8.478 SeizableItem.cpp 1446
8.479 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/SeizableItem.h File Reference 1449
8.480 SeizableItem.h 1450
8.481 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Seize.cpp File Reference 1451
8.482 Seize.cpp 1451

8.483 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Seize.h File Reference . . . 1456
8.484 Seize.h . . . 1456
8.485 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Separate.cpp File Reference . . . 1457
8.486 Separate.cpp . . . 1458
8.487 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Separate.h File Reference . . . 1459
8.488 Separate.h . . . 1459
8.489 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Signal.cpp File Reference . . . 1460
8.490 Signal.cpp . . . 1460
8.491 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Signal.h File Reference . . . 1461
8.492 Signal.h . . . 1462
8.493 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Start.cpp File Reference . . . 1462
8.494 Start.cpp . . . 1462
8.495 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Start.h File Reference . . . 1463
8.496 Start.h . . . 1464
8.497 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Stop.cpp File Reference . . . 1464
8.498 Stop.cpp . . . 1465
8.499 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Stop.h File Reference . . . 1466
8.500 Stop.h . . . 1466
8.501 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Store.cpp File Reference . . . 1467
8.502 Store.cpp . . . 1467
8.503 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Store.h File Reference . . . 1468
8.504 Store.h . . . 1469
8.505 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Submodel.cpp File Reference . . . 1469
8.506 Submodel.cpp . . . 1469
8.507 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Submodel.h File Reference . . . 1470
8.508 Submodel.h . . . 1471
8.509 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Unstore.cpp File Reference . . . 1471
8.510 Unstore.cpp . . . 1472
8.511 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩ S/2019_2022_GenESyS/plugins/components/Unstore.h File Reference . . . 1473
8.512 Unstore.h . . . 1473

8.513 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Write.cpp File Reference . . . 1474
8.514 Write.cpp . . . 1474
8.515 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Write.h File Reference . . . 1476
8.516 Write.h . . . 1477
8.517 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/EntityGroup.cpp File Reference . . . 1478
8.518 EntityGroup.cpp . . . 1479
8.519 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/EntityGroup.h File Reference . . . 1480
8.520 EntityGroup.h . . . 1481
8.521 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Failure.cpp File Reference . . . 1481
8.522 Failure.cpp . . . 1482
8.523 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Failure.h File Reference . . . 1483
8.524 Failure.h . . . 1483
8.525 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/File.cpp File Reference . . . 1484
8.526 File.cpp . . . 1484
8.527 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/File.h File Reference . . . 1485
8.528 File.h . . . 1485
8.529 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Queue.cpp File Reference . . . 1486
8.530 Queue.cpp . . . 1486
8.531 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Queue.h File Reference . . . 1488
8.532 Queue.h . . . 1489
8.533 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Resource.cpp File Reference . . . 1491
8.534 Resource.cpp . . . 1491
8.535 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Resource.h File Reference . . . 1494
8.536 Resource.h . . . 1494
8.537 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Schedule.cpp File Reference . . . 1496
8.538 Schedule.cpp . . . 1496
8.539 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Schedule.h File Reference . . . 1496
8.540 Schedule.h . . . 1497
8.541 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Sequence.cpp File Reference . . . 1497
8.542 Sequence.cpp . . . 1498

8.543 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Sequence.h File Reference . . . 1500
8.544 Sequence.h . . . 1500
8.545 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Set.cpp File Reference . . . 1501
8.546 Set.cpp . . . 1502
8.547 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Set.h File Reference . . . 1503
8.548 Set.h . . . 1504
8.549 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Station.cpp File Reference . . . 1505
8.550 Station.cpp . . . 1505
8.551 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Station.h File Reference . . . 1507
8.552 Station.h . . . 1507
8.553 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Storage.cpp File Reference . . . 1509
8.554 Storage.cpp . . . 1509
8.555 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Storage.h File Reference . . . 1510
8.556 Storage.h . . . 1510
8.557 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Variable.cpp File Reference . . . 1511
8.558 Variable.cpp . . . 1511
8.559 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Variable.h File Reference . . . 1513
8.560 Variable.h . . . 1514
8.561 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tests/testProbabilityDistribution.cpp File Reference . . . 1514
8.561.1 Function Documentation . . . 1515
8.562 testProbabilityDistribution.cpp . . . 1516
8.563 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/HypothesisTester_if.h File Reference . . . 1517
8.563.1 Typedef Documentation . . . 1518
8.564 HypothesisTester_if.h . . . 1518
8.565 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/HypothesisTesterDefaultImpl1.cpp File Reference . . . 1520
8.566 HypothesisTesterDefaultImpl1.cpp . . . 1520
8.567 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/HypothesisTesterDefaultImpl1.h File Reference . . . 1522
8.568 HypothesisTesterDefaultImpl1.h . . . 1523
8.569 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/ProbabilityDistribution.cpp File Reference . . . 1524
8.570 ProbabilityDistribution.cpp . . . 1524
8.571 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/ProbabilityDistribution.h File Reference . . . 1528

8.572 ProbabilityDistribution.h . . . 1529
8.573 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/solver_if.h File Reference . . . 1529
8.574 solver_if.h . . . 1530
8.575 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/SolverDefaultImpl1.cpp File Reference . . . 1530
8.576 SolverDefaultImpl1.cpp . . . 1531
8.577 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/SolverDefaultImpl1.h File Reference . . . 1533
8.578 SolverDefaultImpl1.h . . . 1534
8.579 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/Traits.h File Reference . . . 1534
8.580 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/Traits.h . . . 1535
8.581 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogabout.h File Reference . . . 1536
8.582 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogabout.h . . . 1537
8.583 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogmodelinformation.h File Reference . . . 1538
8.584 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogmodelinformation.h . . . 1539
8.585 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_formPlugins.h File Reference . . . 1541
8.586 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_formPlugins.h . . . 1542

1 Todo List

Member PluginConnectorDummyImpl1::check (const std::string dynamicLibraryFilename)

:To implement

Member PluginInformation::PluginInformation (std::string pluginTypename, StaticLoaderComponentInstance componentloader)

:

Class TraceSimulationProcess

: CLASS NOT FULLY IMPLEMENTED (to be implemented for process analyser)

2 Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Ui **32**

yy **32**

3 Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Assign::Assignment 46

SequenceStep::Assignment 47

Base 53

- **yy::genesyspp_parser::basic_symbol< Base >** 58

yy::genesyspp_parser::basic_symbol< by_state > 58

yy::genesyspp_parser::by_kind 71

- **yy::genesyspp_parser::basic_symbol< by_kind >** 58
  - **yy::genesyspp_parser::symbol_type** 726

Collector_if 82

- **CollectorDatafile_if** 85
  - **CollectorDatafileDefaultImpl1** 87
- **CollectorDefaultImpl1** 92

ComponentManager 95

HypothesisTester_if::ConfidenceInterval 98

ConnectionManager 100

yy::genesyspp_parser::context 105

ModelComponent::DEFAULT_VALUES 122

SourceModelComponent::DEFAULT_VALUES 132

WriteExpression::DEFAULT_VALUES 134

WriteInExpression::DEFAULT_VALUES 135

WriteInText::DEFAULT_VALUES 136

WriteText::DEFAULT_VALUES 137

ElementManager 159

ElementManager_if 164

Event 187

Exact 189

ExperimentManager 197

ExperimentManager_if 200

ExperimentManagerDefaultImpl1 203

ParserManager::GenerateNewParserResult 218

GenesysApplication_if 220

BaseConsoleGenesysApplication 54

AnElectronicAssemblyAndTestSystem 39

Book_Cap02_Example01 67

FullSimulationOfComplexModel 216

GenesysTerminalApp 262

OperatingSystem02 409

OperatingSystem03 411

Smart_AssignWriteSeizes 610

Smart_Delay 613

Smart_Dummy 615

Smart_HoldSignal 617

Smart_ModelInfoModelSimulation 619

Smart_OnEvent 621

Smart_Parser 625

Smart_ParserModelFunctions 627

Smart_Process 629

Smart_ProcessSet 631

Smart_RouteStation 633

Smart_SeizeDelayRelease 635

Smart_SeizeDelayReleaseMany 637

Smart_Sequence 639

GenesysQtGUI 255

GenesysShell_if 257

genesyspp_driver 220

yy::genesyspp_parser 226

HypothesisTester_if 268

HypothesisTesterDefaultImpl1 275

LicenceManager 288

List< T > 292

List< Assign::Assignment * > 292

List< Connection * > 292

List< Counter * > 292

List< double > 292

List< Entity * > 292

List< Event * > 292

List< Model * > 292

List< ModelComponent * > 292

List< ModelElement * > 292

List< ODEfunction * > 292

List< Plugin * > 292

List< SeizableItem * > 292

List< SequenceStep * > 292

List< SimulationControl * > 292

List< simulationEventHandler > 292

List< simulationEventHandlerMethod > 292

List< SimulationExperiment * > 292

List< SimulationResponse * > 292

List< SortedResourceEventHandler * > 292

List< StatisticsCollector * > 292

List< std::map< std::string, double > * > 292

List< std::string > 292

List< traceErrorListener > 292

List< traceErrorListenerMethod > 292

List< traceListener > 292

List< traceListenerMethod > 292

List< traceSimulationListener > 292

List< traceSimulationListenerMethod > 292

List< unsigned int > 292

List< Waiting * > 292

List< WriteText * > 292

yy::location 297

MapKernelAndUI 316
Model 329
ModelChecker_if 339
ModelCheckerDefaultImpl1 341
ModelInfo 357
ModelManager 360
ModelPersistence_if 364
ModelPersistenceDefaultImpl1 365
ModelSimulation 370
ParserManager::NewParser 388
obj_t 389
ODEfunction 391
OnEventManager 397
Parser_if 413
ParserDefaultImpl1 417
ParserDefaultImpl2 419
ParserChangesInformation 414
ParserManager 421
PersistentObject_base 423
ModelElement 349
Attribute 49
Counter 107
Entity 170
EntityGroup 176
EntityType 180
Failure 207
File 211
ModelComponent 344
Access 36
Assign 42
Batch 63
CelularAutomata 78

Decide 115

Delay 140

DropOff 152

Dummy 155

Enter 165

Exit 194

Hold 265

LSODE 301

Leave 283

MarkovChain 319

Match 325

PickStation 424

PickUp 428

Process 460

Record 479

Release 485

Remove 490

Route 505

Search 533

Seize 544

Separate 558

Signal 575

SinkModelComponent 605

Dispose 147

SourceModelComponent 652

Create 111

Start 667

Stop 708

Store 716

Submodel 719

Unstore 790

Write 811

OLD_ODEelement 392

Queue 468

Resource 494

Schedule 530

Sequence 562

Set 569

Station 670

StatisticsCollector 679

Storage 712

Variable 800

SequenceStep 566

Plugin 432

PluginConnector_if 435

PluginConnectorDummyImpl1 437

PluginInformation 439

PluginManager 448

yy::position 451

ProbabilityDistribution 454

QMainWindow 467

MainWindow 306

QueueableItem 475

Sampler_if::RNG_Parameters 504

SamplerBoostImpl::BoostImplRNG_Parameters 70

SamplerDefaultImpl1::DefaultImpl1RNG_Parameters 138

std::runtime_error

yy::genesyspp_parser::syntax_error 729

Sampler_if 512

SamplerBoostImpl 517

SamplerDefaultImpl1 522

ScenarioExperiment_if 529

SeizableItem 537

yy::genesyspp_parser::semantic_type 552

SimulationEvent 581

SimulationExperiment 588

SimulationReporter_if 589

SimulationReporterDefaultImpl1 590

SimulationResponse 594

SimulationControl 579

SimulationScenario 597

Simulator 601

yy::genesyspp_parser::stack< T, S >::slice 608

Solver_if 641

SolverDefaultImpl1 645

SortFile 650

Statistics_if 675

StatisticsDatafile_if 684

StatisticsDatafileDefaultImpl1 687

StatisticsDefaultImpl1 701

yy::genesyspp_parser::symbol_kind 723

HypothesisTester_if::TestResult 730

yy::genesyspp_parser::token 733

TraceEvent 738

TraceErrorEvent 736

TraceSimulationEvent 746

TraceSimulationProcess 749

TraceManager 739

Traits< T > 750

Traits< GenesysApplication_if > 750

Traits< HypothesisTester_if > 751

Traits< Solver_if > 752

TraitsKernel< T > 753

TraitsKernel< Collector_if > 754

TraitsKernel< Model > 754

TraitsKernel< ModelChecker_if > 755

TraitsKernel< ModelComponent > 756

TraitsKernel< ModelElement > 757

TraitsKernel< ModelPersistence_if > 758

TraitsKernel< Parser_if > 759

TraitsKernel< PluginConnector_if > 759

TraitsKernel< Sampler_if > 760

TraitsKernel< SimulationReporter_if > 761

TraitsKernel< Statistics_if > 762

TraitsKernel< StatisticsDatafile_if > 763

Ui_DialogAbout 764

Ui::DialogAbout 145

Ui_DialogModelInformation 767

Ui::DialogModelInformation 146

Ui_formPlugins 771

Ui::formPlugins 215

Ui_MainWindow 773

Ui::MainWindow 315

Ui::MainWindow 315

Util 794

Waiting 806

WaitingResource 809

WriteText 821

WriteExpression 817

WriteInExpression 818

WriteInText 820

vector< stack_symbol_type > ??

4 Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Access 36

AnElectronicAssemblyAndTestSystem 39

Assign 42

Assign::Assignment 46

SequenceStep::Assignment 47

Attribute 49

Base 53

BaseConsoleGenesysApplication 54

yy::genesyspp_parser::basic_symbol< Base > 58

Batch 63

Book_Cap02_Example01 67

SamplerBoostImpl::BoostImplRNG_Parameters 70

yy::genesyspp_parser::by_kind
Type access provider for token (enum) based symbols 71

CelularAutomata 78

Collector_if 82

CollectorDatafile_if 85

CollectorDatafileDefaultImpl1 87

CollectorDefaultImpl1 92

ComponentManager 95

HypothesisTester_if::ConfidenceInterval 98

ConnectionManager 100

yy::genesyspp_parser::context 105

Counter 107

Create 111

Decide 115

ModelComponent::DEFAULT_VALUES 122

SourceModelComponent::DEFAULT_VALUES 132

WriteExpression::DEFAULT_VALUES 134

WriteInExpression::DEFAULT_VALUES 135

WriteInText::DEFAULT_VALUES 136

WriteText::DEFAULT_VALUES 137

SamplerDefaultImpl1::DefaultImpl1RNG_Parameters 138

Delay 140

Ui::DialogAbout 145

Ui::DialogModelInformation 146

Dispose 147

DropOff 152

Dummy 155

ElementManager 159

ElementManager_if 164

Enter 165

Entity 170

EntityGroup 176

EntityType 180

Event 187

Exact 189

Exit 194

ExperimentManager 197

ExperimentManager_if 200

ExperimentManagerDefaultImpl1 203

Failure 207

File 211

Ui::formPlugins 215

FullSimulationOfComplexModel 216

ParserManager::GenerateNewParserResult 218

GenesysApplication_if 220

genesyspp_driver 220

yy::genesyspp_parser
A Bison parser 226

GenesysQtGUI 255

GenesysShell_if 257

GenesysTerminalApp 262

Hold 265

HypothesisTester_if 268

HypothesisTesterDefaultImpl1 275

Leave 283

LicenceManager 288

List< T > 292

yy::location
Two points in a source file 297

LSODE 301

MainWindow 306

Ui::MainWindow 315

MapKernelAndUI 316

MarkovChain 319

Match 325

Model 329

ModelChecker_if 339

ModelCheckerDefaultImpl1 341

ModelComponent 344

ModelElement 349

ModelInfo 357

ModelManager 360

ModelPersistence_if 364

ModelPersistenceDefaultImpl1 365

ModelSimulation 370

ParserManager::NewParser 388

obj_t 389

ODEfunction 391

OLD_ODEelement 392

OnEventManager 397

OperatingSystem02 409

OperatingSystem03 411

Parser_if 413

ParserChangesInformation 414

ParserDefaultImpl1 417

ParserDefaultImpl2 419

ParserManager 421

PersistentObject_base 423

PickStation 424

PickUp 428

Plugin 432

PluginConnector_if 435

PluginConnectorDummyImpl1 437

PluginInformation 439

PluginManager 448

yy::position
A point in a source file 451

ProbabilityDistribution 454

Process 460

QMainWindow 467

Queue 468

QueueableItem 475

Record 479

Release 485

Remove 490

Resource 494

Sampler_if::RNG_Parameters 504

Route 505

Sampler_if 512

SamplerBoostImpl 517

SamplerDefaultImpl1 522

ScenarioExperiment_if 529

Schedule 530

Search 533

SeizableItem 537

Seize 544

yy::genesyspp_parser::semantic_type 552

Separate 558

Sequence 562

SequenceStep 566

Set 569

Signal 575

SimulationControl 579

SimulationEvent 581

SimulationExperiment 588

SimulationReporter_if 589

SimulationReporterDefaultImpl1 590

SimulationResponse 594

SimulationScenario 597

Simulator 601

SinkModelComponent 605

yy::genesyspp_parser::stack< T, S >::slice
Present a slice of the top of a stack 608

Smart_AssignWriteSeizes 610

Smart_Delay 613

Smart_Dummy 615

Smart_HoldSignal 617

Smart_ModelInfoModelSimulation 619

Smart_OnEvent 621

Smart_Parser 625

Smart_ParserModelFunctions 627

Smart_Process 629

Smart_ProcessSet 631

Smart_RouteStation 633

Smart_SeizeDelayRelease 635

Smart_SeizeDelayReleaseMany 637

Smart_Sequence 639

Solver_if 641

SolverDefaultImpl1 645

SortFile 650

SourceModelComponent 652

Start 667

Station 670

Statistics_if 675

StatisticsCollector 679

StatisticsDatafile_if 684

StatisticsDatafileDefaultImpl1 687

StatisticsDefaultImpl1 701

Stop 708

Storage 712

Store 716

Submodel 719

yy::genesyspp_parser::symbol_kind
Symbol kinds 723

yy::genesyspp_parser::symbol_type
"External" symbols: returned by the scanner 726

yy::genesyspp_parser::syntax_error
Syntax errors thrown from user actions 729

HypothesisTester_if::TestResult 730

yy::genesyspp_parser::token
Token kinds 733

TraceErrorEvent 736

TraceEvent 738

TraceManager 739

TraceSimulationEvent 746

TraceSimulationProcess 749

Traits< T > 750

Traits< GenesysApplication_if > 750

Traits< HypothesisTester_if > 751

Traits< Solver_if > 752

TraitsKernel< T > 753

TraitsKernel< Collector_if > 754

TraitsKernel< Model > 754

TraitsKernel< ModelChecker_if > 755

TraitsKernel< ModelComponent > 756

TraitsKernel< ModelElement > 757

TraitsKernel< ModelPersistence_if > 758

TraitsKernel< Parser_if > 759

TraitsKernel< PluginConnector_if > 759

TraitsKernel< Sampler_if > 760

TraitsKernel< SimulationReporter_if > 761

TraitsKernel< Statistics_if > 762

TraitsKernel< StatisticsDatafile_if > 763

Ui_DialogAbout 764

Ui_DialogModelInformation 767

Ui_formPlugins 771

Ui_MainWindow 773

Unstore 790

Util 794

Variable 800

Waiting 806

WaitingResource 809

Write 811

WriteExpression 817

WriteInExpression 818

WriteInText 820

WriteText 821

5 File Index

5.1 File List

Here is a list of all files with brief descriptions:

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/.dep.inc 826

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/main.cpp 1360

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/Traits.h 1534

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogabout.h 1536

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogmodelinformation.h 1538

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_formPlugins.h 1541

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_mainwindow.h 850

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/BaseConsoleGenesysApplication.cpp 826

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/BaseConsoleGenesysApplication.h 829

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/GenesysApplication_if.h 830

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/GenesysQtGUI.cpp 831

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/GenesysQtGUI.h 831

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mapkernelandui.cpp 860

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mapkernelandui.h 861

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.cpp 832

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.h 838

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mainWindow/ui_mainwindow.h 840

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/terminal/GenesysShell_if.h 862

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/terminal/GenesysTerminalApp.cpp 863

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/terminal/GenesysTerminalApp.h 865

html/dynsections.js ??

html/jquery.js ??

html/menu.js ??

html/menudata.js ??

html/search/search.js ??

html/search/searchdata.js ??

user/html/dynsections.js ??

user/html/jquery.js ??

user/html/menu.js ??

user/html/menudata.js ??

user/html/navtree.js ??

user/html/navtreedata.js ??

user/html/navtreeindex0.js ??

user/html/resize.js ??

user/html/search/search.js ??

user/html/search/searchdata.js ??

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/book/Book_Cap02_Example01.cpp 894

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/book/Book_Cap02_Example01.h 895

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_AssignWriteSeizes.cpp 896

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_AssignWriteSeizes.h 898

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Delay.cpp 899

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Delay.h 900

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Dummy.cpp 901

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Dummy.h 903

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_HoldSignal.cpp 904

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_HoldSignal.h 905

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_ModelInfoModelSimulation.cpp 906

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_ModelInfoModelSimulation.h 907

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_OnEvent.cpp 909

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_OnEvent.h 910

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Parser.cpp 912

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Parser.h 913

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.cpp 914

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.h 915

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Process.cpp 917

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Process.h 918

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_ProcessSet.cpp 919

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_ProcessSet.h 920

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_RouteStation.cpp 921

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_RouteStation.h 923

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayRelease.cpp 924

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayRelease.h 925

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.cpp 926

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.h 928

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Sequence.cpp 929

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/smarts/Smart_Sequence.h 931

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/AnElectronicAssemblyAndTestSystem.cpp 932

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/AnElectronicAssemblyAndTestSystem.h 934

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/FullSimulationOfComplexModel.cpp 936

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/FullSimulationOfComplexModel.h 938

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/OperatingSystem02.cpp 939

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/OperatingSystem02.h 941

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/OperatingSystem03.cpp 942

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/examples/teaching/OperatingSystem03.h 946

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/TraitsKernel.h 1343

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Attribute.cpp 948

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Attribute.h 949

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ComponentManager.cpp 950

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ComponentManager.h 952

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ConnectionManager.cpp 953

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ConnectionManager.h 954

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Counter.cpp 956

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Counter.h 958

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/DefineGetterSetter.h 959

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ElementManager.cpp 965

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ElementManager.h 968

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ElementManager_if.h 969

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Entity.cpp 969

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Entity.h 972

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/EntityType.cpp 973

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/EntityType.h 976

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Event.cpp 977

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Event.h 979

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ExperimentManager.cpp 980

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ExperimentManager.h 981

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.cpp 983

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.h 984

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ExperimetManager_if.h 986

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/LicenceManager.cpp 987

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/LicenceManager.h 988

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Model.cpp 990

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Model.h 995

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelChecker_if.h 997

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.cpp 998

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.h 1002

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelComponent.cpp 1003

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelComponent.h 1005

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelElement.cpp 1007

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelElement.h 1011

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelInfo.cpp 1012

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelInfo.h 1014

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelManager.cpp 1015

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelManager.h 1017

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelPersistence_if.h 1018

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelPersistenceDefaultImpl1.cpp 1019

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelPersistenceDefaultImpl1.h 1025

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelSimulation.cpp 1026

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ModelSimulation.h 1035

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/OnEventManager.cpp 1038

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/OnEventManager.h 1041

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Parser_if.h 1045

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserChangesInformation.cpp 1250

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserChangesInformation.h 1250

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl1.cpp 1252

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl1.h 1253

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl2.cpp 1254

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl2.h 1255

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserManager.cpp 1257

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ParserManager.h 1257

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PersistentObject_base.h 1259

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Plugin.cpp 1260

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Plugin.h 1261

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginConnector_if.h 1262

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.cpp 1263

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.h 1266

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginInformation.cpp 1267

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginInformation.h 1270

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginManager.cpp 1273

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/PluginManager.h 1275

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/ScenarioExperiment_if.h 1276

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationControl.cpp 1277

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationControl.h 1277

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationExperiment.cpp 1278

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationExperiment.h 1279

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationReporter_if.h 1280

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.cpp 1281

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.h 1285

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationResponse.cpp 1286

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationResponse.h 1287

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationScenario.cpp 1288

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SimulationScenario.h 1290

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Simulator.cpp 1291

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/Simulator.h 1293

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SinkModelComponent.cpp 1295

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SinkModelComponent.h 1296

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SourceModelComponent.cpp 1297

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/SourceModelComponent.h 1299

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/StatisticsCollector.cpp 1301

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/StatisticsCollector.h 1303

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/TraceManager.cpp 1304

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/TraceManager.h 1306

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-driver.cpp 1046

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-driver.h 1048

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-scanner.cpp 1050

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.cpp 1154

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h 1191

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/location.hh 1230

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.cpp 1235

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.h 1236

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/position.hh 1249

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/stack.hh 1249

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/parser/bisonparser.yy 1237

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/parser/lexerparser.ll 1244

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/statistics/Collector_if.h 1310

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/kernel/statistics/CollectorDatafile_if.h 1312

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.cpp 1313

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.h 1315

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.cpp 1316

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.h 1318

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Sampler_if.h 1319

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerBoostImpl.cpp 1320

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerBoostImpl.h 1321

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.cpp 1322

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.h 1326

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SorttFile.cpp 1327

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SorttFile.h 1329

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Statistics_if.h 1330

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDataFile_if.h 1331

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.cpp 1333

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.h 1337

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDefaultImpl1.cpp 1339

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDefaultImpl1.h 1341

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/util/Exact.h 1345

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/util/List.h 1347

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/util/Util.cpp 1350

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/util/Util.h 1353

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/nbproject/private/c_standard_headers_indexer.c 1361

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/nbproject/private/cpp_standard_headers_indexer.cpp 1363

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Access.cpp 1366

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Access.h 1367

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Assign.cpp 1368

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Assign.h 1370

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Batch.cpp 1371

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Batch.h 1373

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/CellularAutomata.cpp 1374

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/CellularAutomata.h 1375

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Create.cpp 1376

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Create.h 1378

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Decide.cpp 1380

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Decide.h 1382

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Delay.cpp 1383

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Delay.h 1385

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Dispose.cpp 1386

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Dispose.h 1388

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/DropOff.cpp 1389

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/DropOff.h 1390

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Dummy.cpp 1391

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Dummy.h 1393

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Enter.cpp 1394

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Enter.h 1395

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Exit.cpp 1397

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Exit.h 1398

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Hold.cpp 1399

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Hold.h 1400

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Leave.cpp 1401

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Leave.h 1403

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/LSODE.cpp 1404

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/LSODE.h 1406

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/MarkovChain.cpp 1408

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/MarkovChain.h 1410

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Match.cpp 1411

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Match.h 1413

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/OLD_ODEelement.cpp 1414

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/OLD_ODEelement.h 1415

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/PickStation.cpp 1417

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/PickStation.h 1418

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/PickUp.cpp 1419

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/PickUp.h 1420

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Process.cpp 1421

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Process.h 1424

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/QueueableItem.cpp 1425

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/QueueableItem.h 1428

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Record.cpp 1429

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Record.h 1432

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Release.cpp 1433

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Release.h 1435

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Remove.cpp 1437

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Remove.h 1438

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Route.cpp 1439

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Route.h 1442

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Search.cpp 1444

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Search.h 1445

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/SeizableItem.cpp 1446

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/SeizableItem.h 1449

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Seize.cpp 1451

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Seize.h 1456

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Separate.cpp 1457

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Separate.h 1459

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/components/Signal.cpp 1460

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Signal.h 1461

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Start.cpp 1462

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Start.h 1463

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Stop.cpp 1464

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Stop.h 1466

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Store.cpp 1467

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Store.h 1468

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Submodel.cpp 1469

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Submodel.h 1470

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Unstore.cpp 1471

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Unstore.h 1473

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Write.cpp 1474

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Write.h 1476

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/EntityGroup.cpp 1478

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/EntityGroup.h 1480

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Failure.cpp 1481

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Failure.h 1483

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/File.cpp 1484

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/File.h 1485

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Queue.cpp 1486

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Queue.h 1488

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Resource.cpp 1491

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Resource.h 1494

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Schedule.cpp 1496

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Schedule.h 1496

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Sequence.cpp 1497

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Sequence.h 1500

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Set.cpp 1501

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Set.h 1503

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Station.cpp 1505

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Station.h 1507

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Storage.cpp 1509

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Storage.h 1510

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Variable.cpp 1511

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/plugins/elements/Variable.h 1513

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tests/testProbabilityDistribution.cpp 1514

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/HypothesisTester_if.h 1517

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/HypothesisTesterDefaultImpl1.cpp 1520

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/HypothesisTesterDefaultImpl1.h 1522

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/ProbabilityDistribution.cpp 1524

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/ProbabilityDistribution.h 1528

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/solver_if.h 1529

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/SolverDefaultImpl1.cpp 1530

/home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESy↩
S/2019_2022_GenESyS/tools/SolverDefaultImpl1.h 1533

6 Namespace Documentation

6.1 Ui Namespace Reference

Classes

- class MainWindow
- class DialogAbout
- class DialogModelInformation
- class formPlugins

6.2 yy Namespace Reference

Classes

- class genesyspp_parser

  A Bison parser.
- class position

  A point in a source file.
- class location

  Two points in a source file.

Functions

- position & operator+= (position &res, position::counter_type width)

  Add width columns, in place.
- position operator+ (position res, position::counter_type width)

  Add width columns.
- position & operator-= (position &res, position::counter_type width)

  Subtract width columns, in place.
- position operator- (position res, position::counter_type width)

  Subtract width columns.
- template<typename YYChar >
  std::basic_ostream< YYChar > & operator<< (std::basic_ostream< YYChar > &ostr, const position &pos)

  Intercept output stream redirection.
- location & operator+= (location &res, const location &end)

  Join two locations, in place.
- location operator+ (location res, const location &end)

  Join two locations.
- location & operator+= (location &res, location::counter_type width)

  Add width columns to the end position, in place.
- location operator+ (location res, location::counter_type width)

Add width columns to the end position.

- location & operator-= (location &res, location::counter_type width)

  Subtract width columns to the end position, in place.
- location operator- (location res, location::counter_type width)

  Subtract width columns to the end position.
- template<typename YYChar >
  std::basic_ostream< YYChar > & operator<< (std::basic_ostream< YYChar > &ostr, const location &loc)

  Intercept output stream redirection.

6.2.1 Function Documentation

6.2.1.1 operator+() [1/3]

```
location yy::operator+ (
    location res,
    const location & end ) [inline]
```

Join two locations.

Definition at line 243 of file location.hh.

6.2.1.2 operator+() [2/3]

```
location yy::operator+ (
    location res,
    location::counter_type width ) [inline]
```

Add *width* columns to the end position.

Definition at line 258 of file location.hh.

6.2.1.3 operator+() [3/3]

```
position yy::operator+ (
    position res,
    position::counter_type width ) [inline]
```

Add *width* columns.

Definition at line 132 of file location.hh.

6.2.1.4 operator+=() [1/3]

```
location& yy::operator+= (
    location & res,
    const location & end ) [inline]
```

Join two locations, in place.

Definition at line 235 of file location.hh.

6.2.1.5 operator+=() [2/3]

```
location& yy::operator+= (
    location & res,
    location::counter_type width ) [inline]
```

Add *width* columns to the end position, in place.

Definition at line 250 of file location.hh.

Here is the call graph for this function:

6.2.1.6 operator+=() [3/3]

```
position& yy::operator+= (
    position & res,
    position::counter_type width ) [inline]
```

Add *width* columns, in place.

Definition at line 124 of file location.hh.

Here is the call graph for this function:

6.2.1.7 operator-() [1/2]

```
location yy::operator- (
    location res,
    location::counter_type width ) [inline]
```

Subtract *width* columns to the end position.

Definition at line 272 of file location.hh.

6.2.1.8 operator-() [2/2]

```
position yy::operator- (
    position res,
    position::counter_type width ) [inline]
```

Subtract *width* columns.

Definition at line 146 of file location.hh.

6.2.1.9 operator-=() [1/2]

```
location& yy::operator-= (
    location & res,
    location::counter_type width ) [inline]
```

Subtract *width* columns to the end position, in place.

Definition at line 265 of file location.hh.

6.2.1.10 operator-=() [2/2]

```
position& yy::operator-= (
    position & res,
    position::counter_type width ) [inline]
```

Subtract *width* columns, in place.

Definition at line 139 of file location.hh.

6.2.1.11 operator<<() [1/2]

```
template<typename YYChar >
std::basic_ostream<YYChar>& yy::operator<< (
    std::basic_ostream< YYChar > & ostr,
    const location & loc )
```

Intercept output stream redirection.

Parameters

ostr	the destination output stream
loc	a reference to the location to redirect

Avoid duplicate information.

Definition at line 272 of file location.hh.

6.2.1.12 operator<<() [2/2]

```
template<typename YYChar >
std::basic_ostream<YYChar>& yy::operator<< (
```

```
std::basic_ostream< YYChar > & ostr,
const position & pos )
```

Intercept output stream redirection.

Parameters

ostr	the destination output stream
pos	a reference to the position to redirect

Definition at line 146 of file location.hh.

7 Class Documentation

7.1 Access Class Reference

```
#include <Access.h>
```

Inheritance diagram for Access:

Collaboration diagram for Access:

Public Member Functions

- Access (Model ∗model, std::string name="")
- virtual ∼Access ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.1.1 Detailed Description

Access module DESCRIPTION The Access module allocates one or more cells of a conveyor to an entity for movement from one station to another. Once the entity has control of the cells on the conveyor, it may then be conveyed to the next station. When an entity arrives at an Access module, it will wait until the appropriate number of contiguous cells on the conveyor are empty and aligned with the entity's station location. TYPICAL USES Parts accessing a conveyor to be sent to a paint booth Glass accessing a conveyor to be transferred to a cutting station PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Conveyor Name Name of the conveyor that the entity desires.

of Cells Number of contiguous conveyor cells the entity requires for

movement on the conveyor. Queue Type Determines the type of queue used to hold the entities, either an individual Queue, a queue Set, and Internal queue or an Attribute or Expression that evaluate to the queue name. Queue Name Name of the queue that will hold the entity until it accesses the conveyor. Set Name Name of the set of queues. Set Index Defines the index into the queue set. Note that this is the index into the set and not the name of the queue in the set. For example, the only valid entries for a queue set containing three members is an expression that evaluates to 1, 2, or 3. Attribute Name Defines the name of the attribute that stores the queue name to which entities will reside. Expression Defines the name of the expression that stores the queue name to which entities will reside.

Definition at line 53 of file Access.h.

7.1.2 Constructor & Destructor Documentation

7.1.2.1 Access()

```
Access::Access (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Access.cpp.

Here is the caller graph for this function:

7.1.2.2 ~Access()

```
virtual Access::~Access ( ) [virtual], [default]
```

7.1.3 Member Function Documentation

7.1.3.1 _check()

```
bool Access::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Access.cpp.

7.1.3.2 _execute()

```
void Access::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Access.cpp.

Here is the call graph for this function:

7.1.3.3 _initBetweenReplications()

```
void Access::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 48 of file Access.cpp.

7.1.3.4 _loadInstance()

```
bool Access::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 40 of file Access.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.1.3.5 _saveInstance() `std::map< std::string, std::string > * Access::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Access.cpp.

Here is the call graph for this function:

7.1.3.6 GetPluginInformation() `PluginInformation * Access::GetPluginInformation ( ) [static]`

Definition at line 64 of file Access.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.1.3.7 LoadInstance()

```
ModelComponent * Access::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file Access.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.1.3.8 show()

```
std::string Access::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 21 of file Access.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Access.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Access.cpp

7.2 AnElectronicAssemblyAndTestSystem Class Reference

```
#include <AnElectronicAssemblyAndTestSystem.h>
```

Inheritance diagram for AnElectronicAssemblyAndTestSystem:

Collaboration diagram for AnElectronicAssemblyAndTestSystem:

Public Member Functions

- AnElectronicAssemblyAndTestSystem ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.2.1 Detailed Description

Definition at line 18 of file AnElectronicAssemblyAndTestSystem.h.

7.2.2 Constructor & Destructor Documentation

7.2.2.1 AnElectronicAssemblyAndTestSystem() `AnElectronicAssemblyAndTestSystem::AnElectronic↩ AssemblyAndTestSystem ( )`

Definition at line 32 of file AnElectronicAssemblyAndTestSystem.cpp.

7.2.3 Member Function Documentation

7.2.3.1 main()

```
int AnElectronicAssemblyAndTestSystem::main (
    int argc,
    char ** argv ) [virtual]
```

Implements BaseConsoleGenesysApplication.

Definition at line 35 of file AnElectronicAssemblyAndTestSystem.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/AnElectronicAssemblyAndTestSystem.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/AnElectronicAssemblyAndTestSystem.cpp

7.3 Assign Class Reference

```
#include <Assign.h>
```

Inheritance diagram for Assign:

Collaboration diagram for Assign:

Classes

- class Assignment

Public Member Functions

- Assign (Model ∗model, std::string name="")
- virtual ∼Assign ()=default
- virtual std::string show ()
- List< Assignment ∗ > ∗ getAssignments () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.3.1 Detailed Description

Assign module DESCRIPTION This module is used for assigning new values to variables, entity attributes, entity types, entity pictures, or other system variables. Multiple assignments can be made with a single Assign module. TYPICAL USES Accumulate the number of subassemblies added to a part Change an entity's type to represent the customer copy of a multi-page form Establish a customer's priority PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Assignments Specifies the one or more assignments that will be made when an entity executes the module. Type Type of assignment to be made. Other can include system variables, such as resource capacity or simulation end time. Variable Name Name of the variable that will be assigned a new value when an entity enters the module. Applies only when Type is Variable, Variable Array (1D), or Variable Array (2D). Row Specifies the row index for a variable array. Column Specifies the column index for a variable array. Attribute Name Name of the entity attribute that will be assigned a new value when the entity enters the module. Applies only when Type is Attribute. Entity Type New entity type that will be assigned to the entity when the entity enters the module. Applies only when Type is Entity Type. Entity Picture New entity picture that will be assigned to the entity when the entity enters the module. Applies only when Type is Entity Picture. Other Identifies the special system variable that will be assigned a new value when an entity enters the module. Applies only when Type is Other. New Value Assignment value of the attribute, variable, or other system variable. Does not apply when Type is Entity Type or Entity Picture.

Definition at line 58 of file Assign.h.

7.3.2 Constructor & Destructor Documentation

7.3.2.1 Assign()

```
Assign::Assign (
    Model * model,
    std::string name = "" )
```

Definition at line 21 of file Assign.cpp.

Here is the caller graph for this function:

7.3.2.2 ~Assign()

```
virtual Assign::~Assign ( ) [virtual], [default]
```

7.3.3 Member Function Documentation

7.3.3.1 _check()

```
bool Assign::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 97 of file Assign.cpp.

Here is the call graph for this function:

7.3.3.2 _execute()

```
void Assign::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 54 of file Assign.cpp.

Here is the call graph for this function:

7.3.3.3 _initBetweenReplications() `void Assign::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 67 of file Assign.cpp.

7.3.3.4 _loadInstance() `bool Assign::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 70 of file Assign.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.3.3.5 _saveInstance() `std::map< std::string, std::string > * Assign::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 84 of file Assign.cpp.

Here is the call graph for this function:

7.3.3.6 getAssignments() `List< Assign::Assignment * > * Assign::getAssignments ( ) const`

Definition at line 33 of file Assign.cpp.

Here is the caller graph for this function:

7.3.3.7 GetPluginInformation() `PluginInformation * Assign::GetPluginInformation ( ) [static]`

Definition at line 37 of file Assign.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.3.3.8 LoadInstance()

```
ModelComponent * Assign::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 44 of file Assign.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.3.3.9 show()

```
std::string Assign::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 24 of file Assign.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Assign.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Assign.cpp

7.4 Assign::Assignment Class Reference

```
#include <Assign.h>
```

Collaboration diagram for Assign::Assignment:

Public Member Functions

- Assignment (std::string destination, std::string expression)
- void setDestination (std::string _destination)
- std::string getDestination () const
- void setExpression (std::string _expression)
- std::string getExpression () const

7.4.1 Detailed Description

While the assign class allows you to perform multiple assignments, the assignment class defines an assignment itself.

Definition at line 64 of file Assign.h.

7.4.2 Constructor & Destructor Documentation

7.4.2.1 Assignment()

```
Assign::Assignment::Assignment (
    std::string destination,
    std::string expression ) [inline]
```

Definition at line 66 of file Assign.h.

7.4.3 Member Function Documentation

7.4.3.1 getDestination() `std::string Assign::Assignment::getDestination ( ) const [inline]`

Definition at line 75 of file Assign.h.

Here is the caller graph for this function:

7.4.3.2 getExpression() `std::string Assign::Assignment::getExpression ( ) const [inline]`

Definition at line 81 of file Assign.h.

Here is the caller graph for this function:

7.4.3.3 setDestination() `void Assign::Assignment::setDestination ( std::string _destination ) [inline]`

Definition at line 72 of file Assign.h.

7.4.3.4 setExpression()

```
void Assign::Assignment::setExpression (
    std::string _expression ) [inline]
```

Definition at line 78 of file Assign.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Assign.h

7.5 SequenceStep::Assignment Class Reference

```
#include <Sequence.h>
```

Collaboration diagram for SequenceStep::Assignment:

Public Member Functions

- Assignment (std::string destination, std::string expression)
- void setDestination (std::string _destination)
- std::string getDestination () const
- void setExpression (std::string _expression)
- std::string getExpression () const

7.5.1 Detailed Description

Definition at line 25 of file Sequence.h.

7.5.2 Constructor & Destructor Documentation

7.5.2.1 Assignment()

```
SequenceStep::Assignment::Assignment (
    std::string destination,
    std::string expression ) [inline]
```

Definition at line 27 of file Sequence.h.

7.5.3 Member Function Documentation

7.5.3.1 getDestination()

```
std::string SequenceStep::Assignment::getDestination ( ) const [inline]
```

Definition at line 36 of file Sequence.h.

7.5.3.2 getExpression()

```
std::string SequenceStep::Assignment::getExpression ( ) const [inline]
```

Definition at line 42 of file Sequence.h.

7.5.3.3 setDestination()

```
void SequenceStep::Assignment::setDestination (
    std::string _destination ) [inline]
```

Definition at line 33 of file Sequence.h.

7.5.3.4 setExpression()

```
void SequenceStep::Assignment::setExpression (
    std::string _expression ) [inline]
```

Definition at line 39 of file Sequence.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Sequence.h

7.6 Attribute Class Reference

```
#include <Attribute.h>
```

Inheritance diagram for Attribute:

Collaboration diagram for Attribute:

Public Member Functions

- Attribute (Model ∗model, std::string name="")
- virtual ∼Attribute ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.6.1 Detailed Description

Attribute module DESCRIPTION This data module is used to define an attribute's dimension, data type and initial value(s). An attribute is a characteristic of all entities created, but with a specific value that can differ from one entity to another. Attributes can be referenced in other modules (for example, the Decide module), can be reassigned a new value with the Assign module, and can be used in any expression. Attribute values are unique for each entity, as compared to Variables which are global to the simulation module. There are three methods for manually editing the Initial Values of an Attribute module: Using the standard spreadsheet interface. In the module spreadsheet, rightclick on the Initial Values cell and select the Edit via spreadsheet menu item. The values for two-dimensional arrays should be entered one column at a time. Array elements not explicitly assigned are assumed to have the last entered value. Using the module dialog box. In the module spreadsheet, right-click on any cell and select the Edit via dialog menu item. The values for two-dimensional arrays should be entered one column at a time. Array elements not explicitly assigned are assumed to have the last entered value. Using the two-dimensional (2-D) spreadsheet interface. In the module spreadsheet, click on the Initial Values cell. TYPICAL USES Due date of an order (entity) Priority of an order (entity) Color of a part (entity) PROMPTS Prompt Description Name The unique name of the attribute being defined. Rows Number of rows in a one- or two-dimensional attribute. Columns Number of columns in a two-dimensional attribute. Data Type The data type of the values stored in the attribute. Valid types are Real and String. The default type is Real. Initial Values Lists the initial value or values of the attribute. You can assign new values to the attribute by using the Assign module. Initial Value Entity attribute value when entity is created and enters the system.

Definition at line 63 of file Attribute.h.

7.6.2 Constructor & Destructor Documentation

7.6.2.1 Attribute()

```
Attribute::Attribute (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Attribute.cpp.

Here is the caller graph for this function:

7.6.2.2 ~Attribute()

```
virtual Attribute::~Attribute ( ) [virtual], [default]
```

7.6.3 Member Function Documentation

7.6.3.1 _check()

```
bool Attribute::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 50 of file Attribute.cpp.

7.6.3.2 _loadInstance()

```
bool Attribute::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 25 of file Attribute.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.6.3.3 _saveInstance()

```
std::map< std::string, std::string > * Attribute::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 45 of file Attribute.cpp.

Here is the call graph for this function:

7.6.3.4 GetPluginInformation() `PluginInformation * Attribute::GetPluginInformation ( ) [static]`

Definition at line 29 of file Attribute.cpp.

Here is the call graph for this function:

7.6.3.5 LoadInstance() `ModelElement * Attribute::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 35 of file Attribute.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.6.3.6 show() `std::string Attribute::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 21 of file Attribute.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Attribute.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Attribute.cpp

7.7 Base Class Reference

Inheritance diagram for Base:

Collaboration diagram for Base:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.8 BaseConsoleGenesysApplication Class Reference

```
#include <BaseConsoleGenesysApplication.h>
```

Inherits GenesysApplication_if.

Inherited by AnElectronicAssemblyAndTestSystem, Book_Cap02_Example01, FullSimulationOfComplexModel, GenesysTerminalApp, OperatingSystem02, OperatingSystem03, Smart_AssignWriteSeizes, Smart_Delay, Smart_Dummy, Smart_HoldSignal, Smart_ModelInfoModelSimulation, Smart_OnEvent, Smart_Parser, Smart_ParserModelFunctions Smart_Process, Smart_ProcessSet, Smart_RouteStation, Smart_SeizeDelayRelease, Smart_SeizeDelayReleaseMany, and Smart_Sequence.

Collaboration diagram for BaseConsoleGenesysApplication:

Public Member Functions

- BaseConsoleGenesysApplication ()
- virtual ~BaseConsoleGenesysApplication ()=default
- virtual int main (int argc, char **argv)=0
- void setDefaultTraceHandlers (TraceManager *tm)
- void setDefaultEventHandlers (OnEventManager *oem)
- void insertFakePluginsByHand (Simulator *simulator)

Protected Member Functions

- virtual void traceHandler (TraceEvent e)
- virtual void traceErrorHandler (TraceErrorEvent e)
- virtual void traceReportHandler (TraceEvent e)
- virtual void traceSimulationHandler (TraceSimulationEvent e)

- virtual void onBreakpointHandler (SimulationEvent *re)
- virtual void onEntityCreateHandler (SimulationEvent *re)
- virtual void onEntityMoveHandler (SimulationEvent *re)
- virtual void onSimulationStartHandler (SimulationEvent *re)
- virtual void onReplicationStepHandler (SimulationEvent *re)
- virtual void onReplicationStartHandler (SimulationEvent *re)
- virtual void onProcessEventHandler (SimulationEvent *re)
- virtual void onReplicationEndHandler (SimulationEvent *re)
- virtual void onSimulationEndHandler (SimulationEvent *re)
- virtual void onSimulationPausedHandler (SimulationEvent *re)
- virtual void onSimulationResumeHandler (SimulationEvent *re)
- virtual void onEntityRemoveHandler (SimulationEvent *re)

7.8.1 Detailed Description

Definition at line 21 of file BaseConsoleGenesysApplication.h.

7.8.2 Constructor & Destructor Documentation

7.8.2.1 BaseConsoleGenesysApplication() `BaseConsoleGenesysApplication::BaseConsoleGenesys↩ Application ( )`

Definition at line 19 of file BaseConsoleGenesysApplication.cpp.

7.8.2.2 ~BaseConsoleGenesysApplication() `virtual BaseConsoleGenesysApplication::~BaseConsole↩ GenesysApplication ( ) [virtual], [default]`

7.8.3 Member Function Documentation

7.8.3.1 insertFakePluginsByHand() `void BaseConsoleGenesysApplication::insertFakePluginsByHand ( Simulator * simulator )`

Definition at line 114 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.2 main()

```
virtual int BaseConsoleGenesysApplication::main (
    int argc,
    char ** argv )  [pure virtual]
```

Implements GenesysApplication_if.

Implemented in OperatingSystem03, OperatingSystem02, FullSimulationOfComplexModel, AnElectronicAssemblyAndTestSystem, Smart_Sequence, Smart_SeizeDelayReleaseMany, Smart_SeizeDelayRelease, Smart_RouteStation, Smart_ProcessSet, Smart_Process, Smart_ParserModelFunctions, Smart_Parser, Smart_OnEvent, Smart_ModelInfoModelSimulation,

Smart_HoldSignal, Smart_Dummy, Smart_Delay, Smart_AssignWriteSeizes, Book_Cap02_Example01, and GenesysTerminalApp.

7.8.3.3 onBreakpointHandler() `void BaseConsoleGenesysApplication::onBreakpointHandler ( SimulationEvent * re ) [protected], [virtual]`

Reimplemented in Smart_OnEvent.

Definition at line 44 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.4 onEntityCreateHandler() `void BaseConsoleGenesysApplication::onEntityCreateHandler ( SimulationEvent * re ) [protected], [virtual]`

Reimplemented in Smart_OnEvent.

Definition at line 48 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.5 onEntityMoveHandler()

`void BaseConsoleGenesysApplication::onEntityMoveHandler ( SimulationEvent * re ) [protected], [virtual]`

Reimplemented in Smart_OnEvent.

Definition at line 52 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.6 onEntityRemoveHandler()

`void BaseConsoleGenesysApplication::onEntityRemoveHandler ( SimulationEvent * re ) [protected], [virtual]`

Reimplemented in Smart_OnEvent.

Definition at line 56 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.7 onProcessEventHandler() `void BaseConsoleGenesysApplication::onProcessEventHandler ( SimulationEvent * re ) [protected], [virtual]`

Reimplemented in Smart_OnEvent.

Definition at line 60 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.8 onReplicationEndHandler()

```
void BaseConsoleGenesysApplication::onReplicationEndHandler (
        SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 64 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.9 onReplicationStartHandler()

```
void BaseConsoleGenesysApplication::onReplicationStart↩
Handler (
        SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 68 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.10 onReplicationStepHandler()

```
void BaseConsoleGenesysApplication::onReplicationStepHandler (
        SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 72 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.11 onSimulationEndHandler()

```
void BaseConsoleGenesysApplication::onSimulationEndHandler (
        SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 80 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.12 onSimulationPausedHandler()

```
void BaseConsoleGenesysApplication::onSimulationPausedHandler (
            SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 84 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.13 onSimulationResumeHandler()

```
void BaseConsoleGenesysApplication::onSimulationResumeHandler (
            SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 88 of file BaseConsoleGenesysApplication.cpp.

Here is the caller graph for this function:

7.8.3.14 onSimulationStartHandler()

```
void BaseConsoleGenesysApplication::onSimulationStartHandler (
        SimulationEvent * re ) [protected], [virtual]
```

Reimplemented in Smart_OnEvent.

Definition at line 76 of file BaseConsoleGenesysApplication.cpp.

Here is the caller graph for this function:

7.8.3.15 setDefaultEventHandlers()

```
void BaseConsoleGenesysApplication::setDefaultEventHandlers (
        OnEventManager * oem )
```

Definition at line 92 of file BaseConsoleGenesysApplication.cpp.

7.8.3.16 setDefaultTraceHandlers()

```
void BaseConsoleGenesysApplication::setDefaultTraceHandlers (
        TraceManager * tm )
```

Definition at line 107 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.17 traceErrorHandler()

```
void BaseConsoleGenesysApplication::traceErrorHandler (
        TraceErrorEvent e ) [protected], [virtual]
```

Definition at line 29 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.18 traceHandler()

```
void BaseConsoleGenesysApplication::traceHandler (
    TraceEvent e ) [protected], [virtual]
```

Definition at line 25 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.19 traceReportHandler()

```
void BaseConsoleGenesysApplication::traceReportHandler (
    TraceEvent e ) [protected], [virtual]
```

Definition at line 33 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.8.3.20 traceSimulationHandler()

```
void BaseConsoleGenesysApplication::traceSimulationHandler (
    TraceSimulationEvent e ) [protected], [virtual]
```

Definition at line 37 of file BaseConsoleGenesysApplication.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/BaseConsoleGenesysApplication.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/BaseConsoleGenesysApplication.cpp

7.9 yy::genesyspp_parser::basic_symbol< Base > Struct Template Reference

```
#include <GenesysParser.h>
```

Inheritance diagram for yy::genesyspp_parser::basic_symbol< Base >:

Collaboration diagram for yy::genesyspp_parser::basic_symbol< Base >:

Public Types

- typedef Base super_type

  Alias to Base.

Public Member Functions

- basic_symbol ()

Default constructor.

- basic_symbol (const basic_symbol &that)

  Copy constructor.
- basic_symbol (typename Base::kind_type t, const location_type &l)

  Constructors for typed symbols.
- basic_symbol (typename Base::kind_type t, const obj_t &v, const location_type &l)
- ~basic_symbol ()

  Destroy the symbol.
- void clear () YY_NOEXCEPT

  Destroy contents, and record that is empty.
- std::string name () const YY_NOEXCEPT

  The user-facing name of this symbol.
- symbol_kind_type type_get () const YY_NOEXCEPT

  Backward compatibility (Bison 3.6).
- bool empty () const YY_NOEXCEPT

  Whether empty.
- void move (basic_symbol &s)

  Destructive move, s is emptied into this.

Public Attributes

- semantic_type value

  The semantic value.
- location_type location

  The location.

7.9.1 Detailed Description

template<typename Base>
struct yy::genesyspp_parser::basic_symbol< Base >

A complete symbol.

Expects its Base type to provide access to the symbol kind via kind ().

Provide access to semantic value and location.

Definition at line 805 of file GenesysParser.h.

7.9.2 Member Typedef Documentation

7.9.2.1 super_type `template<typename Base >`

`typedef Base yy::genesyspp_parser::basic_symbol< Base >::super_type`

Alias to Base.

Definition at line 808 of file GenesysParser.h.

7.9.3 Constructor & Destructor Documentation

7.9.3.1 basic_symbol() [1/4]

```
template<typename Base >
yy::genesyspp_parser::basic_symbol< Base >::basic_symbol ( ) [inline]
```

Default constructor.

Definition at line 811 of file GenesysParser.h.

7.9.3.2 basic_symbol() [2/4]

```
template<typename Base >
yy::genesyspp_parser::basic_symbol< Base >::basic_symbol (
            const basic_symbol< Base > & that )
```

Copy constructor.

Definition at line 2952 of file GenesysParser.h.

Here is the call graph for this function:

7.9.3.3 basic_symbol() [3/4]

```
template<typename Base >
yy::genesyspp_parser::basic_symbol< Base >::basic_symbol (
            typename Base::kind_type t,
            const location_type & l ) [inline]
```

Constructors for typed symbols.

Definition at line 937 of file GenesysParser.h.

7.9.3.4 basic_symbol() [4/4]

```
template<typename Base >
yy::genesyspp_parser::basic_symbol< Base >::basic_symbol (
            typename Base::kind_type t,
            const obj_t & v,
            const location_type & l ) [inline]
```

Definition at line 950 of file GenesysParser.h.

7.9.3.5 ~basic_symbol() template<typename Base >

`yy::genesyspp_parser::basic_symbol< Base >::~basic_symbol ( ) [inline]`

Destroy the symbol.

Definition at line 958 of file GenesysParser.h.

7.9.4 Member Function Documentation

7.9.4.1 clear() template<typename Base >

`void yy::genesyspp_parser::basic_symbol< Base >::clear ( ) [inline]`

Destroy contents, and record that is empty.

Definition at line 964 of file GenesysParser.h.

Here is the caller graph for this function:

7.9.4.2 empty() template<typename Base >

`bool yy::genesyspp_parser::basic_symbol< Base >::empty ( ) const`

Whether empty.

Definition at line 3071 of file GenesysParser.h.

Here is the caller graph for this function:

7.9.4.3 move()

```
template<typename Base >
void yy::genesyspp_parser::basic_symbol< Base >::move (
            basic_symbol< Base > & s )
```

Destructive move, *s* is emptied into this.

Definition at line 3078 of file GenesysParser.h.

Here is the caller graph for this function:

7.9.4.4 name()

```
template<typename Base >
std::string yy::genesyspp_parser::basic_symbol< Base >::name ( ) const  [inline]
```

The user-facing name of this symbol.

Definition at line 1082 of file GenesysParser.h.

Here is the call graph for this function:

7.9.4.5 type_get()

```
template<typename Base >
genesyspp_parser::symbol_kind_type yy::genesyspp_parser::basic_symbol< Base >::type_get ( )
const
```

Backward compatibility (Bison 3.6).

Definition at line 3064 of file GenesysParser.h.

7.9.5 Member Data Documentation

7.9.5.1 location `template<typename Base >`
`location_type yy::genesyspp_parser::basic_symbol< Base >::location`

The location.

Definition at line 1100 of file GenesysParser.h.

7.9.5.2 value `template<typename Base >`
`semantic_type yy::genesyspp_parser::basic_symbol< Base >::value`

The semantic value.

Definition at line 1097 of file GenesysParser.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.10 Batch Class Reference

```
#include <Batch.h>
```

Inheritance diagram for Batch:

Collaboration diagram for Batch:

Public Member Functions

- Batch (Model ∗model, std::string name="")
- virtual ∼Batch ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.10.1 Detailed Description

Batch module DESCRIPTION This module is intended as the grouping mechanism within the simulation model. Batches can be permanently or temporarily grouped. Temporary batches must later be split using the Separate module. Batches may be made with any specified number of entering entities or may be matched together based on an attribute. Entities arriving at the Batch module are placed in a queue until the required number of entities has accumulated. Once accumulated, a new representative entity is created. TYPICAL USES Collect a number of parts before starting processing Reassemble previously separated copies of a form Bring together a patient and his record before commencing an appointment PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Method of batching entities together. Batch Size Number of entities to be batched. Save Criterion Method for assigning representative entity's user-defined attribute values. Rule Determines how incoming entities will be batched. Any Entity will take the first "Batch Size" number of entities and put them together. By Attribute signifies that the values of the specified attribute must match for entities to be grouped. For example, if Attribute Name is Color, all entities must have the same Color value to be grouped; otherwise, they will wait at the module for additional incoming entities. Attribute Name Name of the attribute whose value must match the value of the other incoming entities in order for a group to be made. Applies only when Rule is By Attribute. Representative Entity The entity type for the representative entity.

Definition at line 53 of file Batch.h.

7.10.2 Constructor & Destructor Documentation

7.10.2.1 Batch()

```
Batch::Batch (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Batch.cpp.

Here is the caller graph for this function:

7.10.2.2 ~Batch()

```
virtual Batch::~Batch ( ) [virtual], [default]
```

7.10.3 Member Function Documentation

7.10.3.1 _check()

```
bool Batch::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Batch.cpp.

7.10.3.2 _execute()

```
void Batch::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Batch.cpp.

Here is the call graph for this function:

7.10.3.3 _initBetweenReplications()

```
void Batch::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 48 of file Batch.cpp.

7.10.3.4 _loadInstance() `bool Batch::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 40 of file Batch.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.10.3.5 _saveInstance() `std::map< std::string, std::string > * Batch::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Batch.cpp.

Here is the call graph for this function:

7.10.3.6 GetPluginInformation() `PluginInformation * Batch::GetPluginInformation ( ) [static]`

Definition at line 64 of file Batch.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.10.3.7 LoadInstance() `ModelComponent * Batch::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 25 of file Batch.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.10.3.8 show() `std::string Batch::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file Batch.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Batch.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Batch.cpp

7.11 Book_Cap02_Example01 Class Reference

```
#include <Book_Cap02_Example01.h>
```

Inheritance diagram for Book_Cap02_Example01:

Collaboration diagram for Book_Cap02_Example01:

Public Member Functions

- Book_Cap02_Example01 ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.11.1 Detailed Description

Definition at line 18 of file Book_Cap02_Example01.h.

7.11.2 Constructor & Destructor Documentation

7.11.2.1 Book_Cap02_Example01() `Book_Cap02_Example01::Book_Cap02_Example01 ( )`

Definition at line 20 of file Book_Cap02_Example01.cpp.

7.11.3 Member Function Documentation

7.11.3.1 main()

```
int Book_Cap02_Example01::main (
    int argc,
    char ** argv ) [virtual]
```

Implements BaseConsoleGenesysApplication.

Definition at line 23 of file Book_Cap02_Example01.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/book/Book_Cap02_Example01.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/book/Book_Cap02_Example01.cpp

7.12 SamplerBoostImpl::BoostImplRNG_Parameters Struct Reference

```
#include <SamplerBoostImpl.h>
```

Inheritance diagram for SamplerBoostImpl::BoostImplRNG_Parameters:

Collaboration diagram for SamplerBoostImpl::BoostImplRNG_Parameters:

Public Attributes

- double param

Additional Inherited Members

7.12.1 Detailed Description

Definition at line 23 of file SamplerBoostImpl.h.

7.12.2 Member Data Documentation

7.12.2.1 param `double SamplerBoostImpl::BoostImplRNG_Parameters::param`

Definition at line 24 of file SamplerBoostImpl.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SamplerBoostImpl.h

7.13 yy::genesyspp_parser::by_kind Struct Reference

Type access provider for token (enum) based symbols.

```
#include <GenesysParser.h>
```

Inheritance diagram for yy::genesyspp_parser::by_kind:

Collaboration diagram for yy::genesyspp_parser::by_kind:

Public Types

- typedef token_kind_type kind_type

  The symbol kind as needed by the constructor.

Public Member Functions

- by_kind ()

  Default constructor.
- by_kind (const by_kind &that)

  Copy constructor.
- by_kind (kind_type t)

  Constructor from (external) token numbers.
- void clear () YY_NOEXCEPT

  Record that this symbol is empty.
- void move (by_kind &that)

  Steal the symbol kind from that.
- symbol_kind_type kind () const YY_NOEXCEPT
- symbol_kind_type type_get () const YY_NOEXCEPT

  Backward compatibility (Bison 3.6).

Public Attributes

- symbol_kind_type kind_

7.13.1 Detailed Description

Type access provider for token (enum) based symbols.

Definition at line 1110 of file GenesysParser.h.

7.13.2 Member Typedef Documentation

7.13.2.1 kind_type `typedef token_kind_type yy::genesyspp_parser::by_kind::kind_type`

The symbol kind as needed by the constructor.

Definition at line 1124 of file GenesysParser.h.

7.13.3 Constructor & Destructor Documentation

7.13.3.1 by_kind() [1/3] `yy::genesyspp_parser::by_kind::by_kind ( ) [inline]`

Default constructor.

Definition at line 3187 of file GenesysParser.h.

7.13.3.2 by_kind() [2/3] `yy::genesyspp_parser::by_kind::by_kind ( const by_kind & that ) [inline]`

Copy constructor.

Definition at line 3201 of file GenesysParser.h.

7.13.3.3 by_kind() [3/3] `yy::genesyspp_parser::by_kind::by_kind ( kind_type t ) [inline]`

Constructor from (external) token numbers.

Definition at line 3206 of file GenesysParser.h.

7.13.4 Member Function Documentation

7.13.4.1 clear() `void yy::genesyspp_parser::by_kind::clear ( ) [inline]`

Record that this symbol is empty.

Definition at line 3212 of file GenesysParser.h.

Here is the caller graph for this function:

7.13.4.2 kind() `genesyspp_parser::symbol_kind_type yy::genesyspp_parser::by_kind::kind ( ) const [inline]`

The (internal) type number (corresponding to *type*). *empty* when empty.

Definition at line 3227 of file GenesysParser.h.

Here is the caller graph for this function:

7.13.4.3 move() `void yy::genesyspp_parser::by_kind::move ( by_kind & that ) [inline]`

Steal the symbol kind from *that*.

Definition at line 3219 of file GenesysParser.h.

Here is the call graph for this function:

7.13.4.4 type_get() genesyspp_parser::symbol_kind_type yy::genesyspp_parser::by_kind::type_get ( ) const [inline]

Backward compatibility (Bison 3.6).

Definition at line 3234 of file GenesysParser.h.

7.13.5 Member Data Documentation

7.13.5.1 kind_ symbol_kind_type yy::genesyspp_parser::by_kind::kind_

The symbol kind. *S_YYEMPTY* when empty.

Definition at line 1144 of file GenesysParser.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.14 CelularAutomata Class Reference

```
#include <CellularAutomata.h>
```

Inheritance diagram for CelularAutomata:

Collaboration diagram for CelularAutomata:

Public Member Functions

- CelularAutomata (Model ∗model, std::string name="")
- virtual ∼CelularAutomata ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.14.1 Detailed Description

Definition at line 21 of file CellularAutomata.h.

7.14.2 Constructor & Destructor Documentation

7.14.2.1 CelularAutomata()

```
CelularAutomata::CelularAutomata (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file CellularAutomata.cpp.

Here is the caller graph for this function:

7.14.2.2 ~CelularAutomata()

```
virtual CelularAutomata::~CelularAutomata ( ) [virtual], [default]
```

7.14.3 Member Function Documentation

7.14.3.1 _check()

```
bool CelularAutomata::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file CellularAutomata.cpp.

7.14.3.2 _execute()

```
void CelularAutomata::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file CellularAutomata.cpp.

Here is the call graph for this function:

7.14.3.3 _initBetweenReplications() `void CelularAutomata::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 47 of file CellularAutomata.cpp.

7.14.3.4 _loadInstance() `bool CelularAutomata::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 39 of file CellularAutomata.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.14.3.5 _saveInstance() `std::map< std::string, std::string > * CelularAutomata::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 50 of file CellularAutomata.cpp.

Here is the call graph for this function:

7.14.3.6 GetPluginInformation() `PluginInformation * CelularAutomata::GetPluginInformation ( ) [static]`

Definition at line 63 of file CellularAutomata.cpp.

Here is the call graph for this function:

7.14.3.7 LoadInstance() `ModelComponent * CelularAutomata::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 24 of file CellularAutomata.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.14.3.8 show() `std::string CelularAutomata::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 20 of file CellularAutomata.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/CellularAutomata.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/CellularAutomata.cpp

7.15 Collector_if Class Reference

```
#include <Collector_if.h>
```

Inheritance diagram for Collector_if:

Collaboration diagram for Collector_if:

Public Member Functions

- virtual void clear ()=0
- virtual void addValue (double value)=0
- virtual double getLastValue ()=0
- virtual unsigned long numElements ()=0
- virtual void setAddValueHandler (CollectorAddValueHandler addValueHandler)=0
- virtual void setClearHandler (CollectorClearHandler clearHandler)=0

7.15.1 Detailed Description

Interface for collecting values of a single stochastic variable. Values collected can be used as base for statistical analysis.

Definition at line 37 of file Collector_if.h.

7.15.2 Member Function Documentation

7.15.2.1 addValue() `virtual void Collector_if::addValue ( double value ) [pure virtual]`

Implemented in CollectorDefaultImpl1, and CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

7.15.2.2 clear() `virtual void Collector_if::clear ( ) [pure virtual]`

Implemented in CollectorDefaultImpl1, and CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

7.15.2.3 getLastValue() `virtual double Collector_if::getLastValue ( ) [pure virtual]`

Implemented in CollectorDefaultImpl1, and CollectorDatafileDefaultImpl1.

7.15.2.4 numElements() `virtual unsigned long Collector_if::numElements ( ) [pure virtual]`

Implemented in CollectorDefaultImpl1, and CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

7.15.2.5 setAddValueHandler() `virtual void Collector_if::setAddValueHandler ( CollectorAddValueHandler addValueHandler ) [pure virtual]`

Implemented in CollectorDefaultImpl1, and CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

7.15.2.6 setClearHandler() `virtual void Collector_if::setClearHandler ( CollectorClearHandler clearHandler ) [pure virtual]`

Implemented in CollectorDefaultImpl1, and CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/Collector_if.h

7.16 CollectorDatafile_if Class Reference

```
#include <CollectorDatafile_if.h>
```

Inheritance diagram for CollectorDatafile_if:

Collaboration diagram for CollectorDatafile_if:

Public Member Functions

- virtual double getValue (unsigned int rank)=0
- virtual void seekFirstValue ()=0
- virtual double getNextValue ()=0
- virtual std::string getDataFilename ()=0
- virtual void setDataFilename (std::string filename)=0

7.16.1 Detailed Description

Interface for collecting values of a stochastic variable that will be stores in a datafile.

Definition at line 22 of file CollectorDatafile_if.h.

7.16.2 Member Function Documentation

7.16.2.1 getDataFilename() `virtual std::string CollectorDatafile_if::getDataFilename ( ) [pure virtual]`

Implemented in CollectorDatafileDefaultImpl1.

7.16.2.2 getNextValue() `virtual double CollectorDatafile_if::getNextValue ( ) [pure virtual]`

Get the next value in the file and advances the pointer

Implemented in CollectorDatafileDefaultImpl1.

7.16.2.3 getValue() `virtual double CollectorDatafile_if::getValue ( unsigned int rank ) [pure virtual]`

Get a value from a specific position

Implemented in CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

7.16.2.4 seekFirstValue() `virtual void CollectorDatafile_if::seekFirstValue ( ) [pure virtual]`

Set the pointer to the first value in the file

Implemented in CollectorDatafileDefaultImpl1.

7.16.2.5 setDataFilename()

```
virtual void CollectorDatafile_if::setDataFilename (
    std::string filename ) [pure virtual]
```

Implemented in CollectorDatafileDefaultImpl1.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/CollectorDatafile_if.h

7.17 CollectorDatafileDefaultImpl1 Class Reference

```
#include <CollectorDatafileDefaultImpl1.h>
```

Inheritance diagram for CollectorDatafileDefaultImpl1:

Collaboration diagram for CollectorDatafileDefaultImpl1:

Public Member Functions

- CollectorDatafileDefaultImpl1 ()
- virtual ~CollectorDatafileDefaultImpl1 ()=default
- void clear ()
- void addValue (double value)
- double getLastValue ()

- unsigned long numElements ()
- double getValue (unsigned int num)
- double getNextValue ()
- void seekFirstValue ()
- std::string getDataFilename ()
- void setDataFilename (std::string filename)
- void setAddValueHandler (CollectorAddValueHandler addValueHandler)
- void setClearHandler (CollectorClearHandler clearHandler)

7.17.1 Detailed Description

Definition at line 21 of file CollectorDatafileDefaultImpl1.h.

7.17.2 Constructor & Destructor Documentation

7.17.2.1 CollectorDatafileDefaultImpl1() `CollectorDatafileDefaultImpl1::CollectorDatafileDefaultImpl1 ( )`

Definition at line 21 of file CollectorDatafileDefaultImpl1.cpp.

7.17.2.2 ~CollectorDatafileDefaultImpl1() `virtual CollectorDatafileDefaultImpl1::~CollectorDatafileDefaultImpl1 ( ) [virtual], [default]`

7.17.3 Member Function Documentation

7.17.3.1 addValue() `void CollectorDatafileDefaultImpl1::addValue ( double value ) [virtual]`

Implements Collector_if.

Definition at line 36 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.2 clear() `void CollectorDatafileDefaultImpl1::clear ( ) [virtual]`

Implements Collector_if.

Definition at line 24 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.3 getDataFilename() `std::string CollectorDatafileDefaultImpl1::getDataFilename ( ) [virtual]`

Implements CollectorDatafile_if.

Definition at line 86 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.4 getLastValue() `double CollectorDatafileDefaultImpl1::getLastValue ( ) [virtual]`

Implements Collector_if.

Definition at line 53 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.5 getNextValue() `double CollectorDatafileDefaultImpl1::getNextValue ( ) [virtual]`

Get the next value in the file and advances the pointer

Implements CollectorDatafile_if.

Definition at line 79 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.6 getValue() `double CollectorDatafileDefaultImpl1::getValue ( unsigned int rank ) [virtual]`

Get a value from a specific position

Implements CollectorDatafile_if.

Definition at line 61 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.7 numElements() `unsigned long CollectorDatafileDefaultImpl1::numElements ( ) [virtual]`

Implements Collector_if.

Definition at line 57 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.8 seekFirstValue() `void CollectorDatafileDefaultImpl1::seekFirstValue ( ) [virtual]`

Set the pointer to the first value in the file

Implements CollectorDatafile_if.

Definition at line 83 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.9 setAddValueHandler()

`void CollectorDatafileDefaultImpl1::setAddValueHandler ( CollectorAddValueHandler addValueHandler ) [virtual]`

Implements Collector_if.

Definition at line 94 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.10 setClearHandler()

`void CollectorDatafileDefaultImpl1::setClearHandler ( CollectorClearHandler clearHandler ) [virtual]`

Implements Collector_if.

Definition at line 98 of file CollectorDatafileDefaultImpl1.cpp.

7.17.3.11 setDataFilename()

`void CollectorDatafileDefaultImpl1::setDataFilename ( std::string filename ) [virtual]`

Implements CollectorDatafile_if.

Definition at line 90 of file CollectorDatafileDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.cpp

7.18 CollectorDefaultImpl1 Class Reference

```
#include <CollectorDefaultImpl1.h>
```

Inheritance diagram for CollectorDefaultImpl1:

Collaboration diagram for CollectorDefaultImpl1:

Public Member Functions

- CollectorDefaultImpl1 ()
- virtual ~CollectorDefaultImpl1 ()=default
- void clear ()
- void addValue (double value)
- double getLastValue ()
- unsigned long numElements ()
- void setAddValueHandler (CollectorAddValueHandler addValueHandler)
- void setClearHandler (CollectorClearHandler clearHandler)

7.18.1 Detailed Description

Definition at line 20 of file CollectorDefaultImpl1.h.

7.18.2 Constructor & Destructor Documentation

7.18.2.1 CollectorDefaultImpl1() `CollectorDefaultImpl1::CollectorDefaultImpl1 ( )`

Definition at line 18 of file CollectorDefaultImpl1.cpp.

7.18.2.2 ~CollectorDefaultImpl1() `virtual CollectorDefaultImpl1::~CollectorDefaultImpl1 ( ) [virtual], [default]`

7.18.3 Member Function Documentation

7.18.3.1 addValue() `void CollectorDefaultImpl1::addValue ( double value ) [virtual]`

Implements Collector_if.

Definition at line 28 of file CollectorDefaultImpl1.cpp.

7.18.3.2 clear() `void CollectorDefaultImpl1::clear ( ) [virtual]`

Implements Collector_if.

Definition at line 21 of file CollectorDefaultImpl1.cpp.

7.18.3.3 getLastValue() `double CollectorDefaultImpl1::getLastValue ( ) [virtual]`

Implements Collector_if.

Definition at line 36 of file CollectorDefaultImpl1.cpp.

7.18.3.4 numElements() `unsigned long CollectorDefaultImpl1::numElements ( ) [virtual]`

Implements Collector_if.

Definition at line 40 of file CollectorDefaultImpl1.cpp.

7.18.3.5 setAddValueHandler() `void CollectorDefaultImpl1::setAddValueHandler ( CollectorAddValueHandler addValueHandler ) [virtual]`

Implements Collector_if.

Definition at line 44 of file CollectorDefaultImpl1.cpp.

7.18.3.6 setClearHandler() `void CollectorDefaultImpl1::setClearHandler ( CollectorClearHandler clearHandler ) [virtual]`

Implements Collector_if.

Definition at line 48 of file CollectorDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.cpp

7.19 ComponentManager Class Reference

```
#include <ComponentManager.h>
```

Collaboration diagram for ComponentManager:

Public Member Functions

- ComponentManager (Model ∗model)
- virtual ∼ComponentManager ()=default
- bool insert (ModelComponent ∗comp)
- void remove (ModelComponent ∗comp)
- ModelComponent ∗ find (std::string name)
- void clear ()
- unsigned int getNumberOfComponents ()
- std::list< ModelComponent ∗ >::iterator begin ()
- std::list< ModelComponent ∗ >::iterator end ()
- ModelComponent ∗ front ()
- ModelComponent ∗ next ()
- bool hasChanged () const
- void setHasChanged (bool _hasChanged)

7.19.1 Detailed Description

ComponentManager allows to insert, access, find, remove and over Models

Definition at line 25 of file ComponentManager.h.

7.19.2 Constructor & Destructor Documentation

7.19.2.1 ComponentManager() `ComponentManager::ComponentManager ( Model * model )`

Components are sorted by ID

Definition at line 20 of file ComponentManager.cpp.

Here is the call graph for this function:

7.19.2.2 ∼ComponentManager() `virtual ComponentManager::∼ComponentManager ( ) [virtual], [default]`

7.19.3 Member Function Documentation

7.19.3.1 begin() `std::list< ModelComponent * >::iterator ComponentManager::begin ( )`

Definition at line 69 of file ComponentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.19.3.2 clear() `void ComponentManager::clear ( )`

Definition at line 54 of file ComponentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.19.3.3 end() `std::list< ModelComponent * >::iterator ComponentManager::end ( )`

Definition at line 73 of file ComponentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.19.3.4 find() `ModelComponent * ComponentManager::find ( std::string name )`

Definition at line 39 of file ComponentManager.cpp.

Here is the call graph for this function:

7.19.3.5 front() `ModelComponent * ComponentManager::front ( )`

Definition at line 77 of file ComponentManager.cpp.

Here is the call graph for this function:

7.19.3.6 getNumberOfComponents() `unsigned int ComponentManager::getNumberOfComponents ( )`

Definition at line 65 of file ComponentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.19.3.7 hasChanged() `bool ComponentManager::hasChanged ( ) const`

Definition at line 85 of file ComponentManager.cpp.

Here is the call graph for this function:

7.19.3.8 insert() `bool ComponentManager::insert ( ModelComponent * comp )`

Definition at line 28 of file ComponentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.19.3.9 next() `ModelComponent * ComponentManager::next ( )`

Definition at line 81 of file ComponentManager.cpp.

Here is the call graph for this function:

7.19.3.10 remove()

```
void ComponentManager::remove (
    ModelComponent * comp )
```

Definition at line 48 of file ComponentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.19.3.11 setHasChanged()

```
void ComponentManager::setHasChanged (
    bool _hasChanged )
```

Definition at line 97 of file ComponentManager.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ComponentManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ComponentManager.cpp

7.20 HypothesisTester_if::ConfidenceInterval Class Reference

```
#include <HypothesisTester_if.h>
```

Collaboration diagram for HypothesisTester_if::ConfidenceInterval:

Public Member Functions

- ConfidenceInterval (double inferiorLimit, double superiorLimit)
- double inferiorLimit ()
- double superiorLimit ()

7.20.1 Detailed Description

Definition at line 28 of file HypothesisTester_if.h.

7.20.2 Constructor & Destructor Documentation

7.20.2.1 ConfidenceInterval()

```
HypothesisTester_if::ConfidenceInterval::ConfidenceInterval (
    double inferiorLimit,
    double superiorLimit ) [inline]
```

Definition at line 30 of file HypothesisTester_if.h.

Here is the call graph for this function:

7.20.3 Member Function Documentation

7.20.3.1 inferiorLimit() `double HypothesisTester_if::ConfidenceInterval::inferiorLimit ( ) [inline]`

Definition at line 34 of file HypothesisTester_if.h.

Here is the caller graph for this function:

7.20.3.2 superiorLimit() `double HypothesisTester_if::ConfidenceInterval::superiorLimit ( ) [inline]`

Definition at line 37 of file HypothesisTester_if.h.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/HypothesisTester_if.h

7.21 ConnectionManager Class Reference

```
#include <ConnectionManager.h>
```

Collaboration diagram for ConnectionManager:

Public Member Functions

- ConnectionManager ()
- virtual ~ConnectionManager ()=default
- unsigned int size ()
- ModelComponent * front ()
- ModelComponent * atRank (unsigned int rank)
- Connection * getFrontConnection ()
- Connection * getConnectionAtRank (unsigned int rank)
- void insert (ModelComponent *component, unsigned int inputNumber=0)
- void insert (Connection *connection)
- void insertAtRank (unsigned int rank, Connection *connection)
- std::list< Connection * > * list () const
- unsigned int getCurrentOutputConnections () const
- void setMaxOutputConnections (unsigned int _maxOutputConnections)
- unsigned int getMaxOutputConnections () const
- void setMinOutputConnections (unsigned int _minOutputConnections)
- unsigned int getMinOutputConnections () const
- unsigned int getCurrentInputConnections () const
- void setMaxInputConnections (unsigned int _maxInputConnections)
- unsigned int getMaxInputConnections () const
- void setMinInputConnections (unsigned int _minInputConnections)
- unsigned int getMinInputConnections () const

7.21.1 Detailed Description

ConnectionManager defines how a ModelComponent is output connected to none, one or more following Model↩ Components. It has a list of nextConnections, where each Connection is a pair, defining the next ModelComponent and an input port on that component (usefull only if the next component has more than one input). The number of the output conection is its rank in the nextConnections list. Min and max number of input and output connectons can be defined.

Definition at line 29 of file ConnectionManager.h.

7.21.2 Constructor & Destructor Documentation

7.21.2.1 ConnectionManager() `ConnectionManager::ConnectionManager ( )`

Definition at line 18 of file ConnectionManager.cpp.

7.21.2.2 ~ConnectionManager() `virtual ConnectionManager::~ConnectionManager ( ) [virtual], [default]`

7.21.3 Member Function Documentation

7.21.3.1 atRank() `ModelComponent * ConnectionManager::atRank ( unsigned int rank )`

DEPRECTED. Use getConnectionAtRank instead

Definition at line 29 of file ConnectionManager.cpp.

Here is the call graph for this function:

7.21.3.2 front() `ModelComponent * ConnectionManager::front ( )`

DEPRECTED. Use frontConnection instead

Definition at line 25 of file ConnectionManager.cpp.

Here is the call graph for this function:

7.21.3.3 getConnectionAtRank() `Connection * ConnectionManager::getConnectionAtRank ( unsigned int rank )`

Definition at line 37 of file ConnectionManager.cpp.

Here is the call graph for this function:

7.21.3.4 getCurrentInputConnections() `unsigned int ConnectionManager::getCurrentInputConnections ( ) const`

Definition at line 90 of file ConnectionManager.cpp.

7.21.3.5 getCurrentOutputConnections() `unsigned int ConnectionManager::getCurrentOutputConnections ( ) const`

Definition at line 66 of file ConnectionManager.cpp.

7.21.3.6 getFrontConnection() `Connection * ConnectionManager::getFrontConnection ( )`

Definition at line 33 of file ConnectionManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.21.3.7 getMaxInputConnections() `unsigned int ConnectionManager::getMaxInputConnections ( ) const`

Definition at line 98 of file ConnectionManager.cpp.

7.21.3.8 getMaxOutputConnections() `unsigned int ConnectionManager::getMaxOutputConnections ( ) const`

Definition at line 74 of file ConnectionManager.cpp.

7.21.3.9 getMinInputConnections() `unsigned int ConnectionManager::getMinInputConnections ( ) const`

Definition at line 106 of file ConnectionManager.cpp.

7.21.3.10 getMinOutputConnections() `unsigned int ConnectionManager::getMinOutputConnections ( ) const`

Definition at line 82 of file ConnectionManager.cpp.

7.21.3.11 insert() [1/2]

```
void ConnectionManager::insert (
    Connection * connection )
```

Definition at line 46 of file ConnectionManager.cpp.

Here is the call graph for this function:

7.21.3.12 insert() [2/2]

```
void ConnectionManager::insert (
    ModelComponent * component,
    unsigned int inputNumber = 0 )
```

Definition at line 41 of file ConnectionManager.cpp.

Here is the caller graph for this function:

7.21.3.13 insertAtRank()

```
void ConnectionManager::insertAtRank (
    unsigned int rank,
    Connection * connection )
```

Definition at line 53 of file ConnectionManager.cpp.

Here is the call graph for this function:

7.21.3.14 list() `std::list< Connection * > * ConnectionManager::list ( ) const`

Definition at line 58 of file ConnectionManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.21.3.15 setMaxInputConnections()

`void ConnectionManager::setMaxInputConnections ( unsigned int _maxInputConnections )`

Definition at line 94 of file ConnectionManager.cpp.

Here is the caller graph for this function:

7.21.3.16 setMaxOutputConnections()

`void ConnectionManager::setMaxOutputConnections ( unsigned int _maxOutputConnections )`

Definition at line 70 of file ConnectionManager.cpp.

Here is the caller graph for this function:

7.21.3.17 setMinInputConnections() `void ConnectionManager::setMinInputConnections ( unsigned int _minInputConnections )`

Definition at line 102 of file ConnectionManager.cpp.

Here is the caller graph for this function:

7.21.3.18 setMinOutputConnections() `void ConnectionManager::setMinOutputConnections ( unsigned int _minOutputConnections )`

Definition at line 78 of file ConnectionManager.cpp.

Here is the caller graph for this function:

7.21.3.19 size() `unsigned int ConnectionManager::size ( )`

Definition at line 21 of file ConnectionManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ConnectionManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ConnectionManager.cpp

7.22 yy::genesyspp_parser::context Class Reference

```
#include <GenesysParser.h>
```

Collaboration diagram for yy::genesyspp_parser::context:

Public Member Functions

- context (const genesyspp_parser &yyparser, const symbol_type &yyla)
- const symbol_type & lookahead () const YY_NOEXCEPT
- symbol_kind_type token () const YY_NOEXCEPT
- const location_type & location () const YY_NOEXCEPT
- int expected_tokens (symbol_kind_type yyarg[ ], int yyargn) const

7.22.1 Detailed Description

Definition at line 2554 of file GenesysParser.h.

7.22.2 Constructor & Destructor Documentation

7.22.2.1 context()

```
yy::genesyspp_parser::context::context (
    const genesyspp_parser & yyparser,
    const symbol_type & yyla )
```

Definition at line 1973 of file GenesysParser.cpp.

7.22.3 Member Function Documentation

7.22.3.1 expected_tokens()

```
int yy::genesyspp_parser::context::expected_tokens (
    symbol_kind_type yyarg[],
    int yyargn ) const
```

Put in YYARG at most YYARGN of the expected tokens, and return the number of tokens stored in YYARG. If YYARG is null, return the number of expected tokens (guaranteed to be less than YYNTOKENS).

Definition at line 1979 of file GenesysParser.cpp.

7.22.3.2 location()

```
const location_type& yy::genesyspp_parser::context::location ( ) const [inline]
```

Definition at line 2560 of file GenesysParser.h.

7.22.3.3 lookahead()

```
const symbol_type& yy::genesyspp_parser::context::lookahead ( ) const [inline]
```

Definition at line 2558 of file GenesysParser.h.

7.22.3.4 token()

```
symbol_kind_type yy::genesyspp_parser::context::token ( ) const [inline]
```

Definition at line 2559 of file GenesysParser.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.cpp

7.23 Counter Class Reference

```
#include <Counter.h>
```

Inheritance diagram for Counter:

Collaboration diagram for Counter:

Public Member Functions

- Counter (Model ∗model, std::string name="", ModelElement ∗parent=nullptr)
- virtual ∼Counter ()=default
- virtual std::string show ()
- void clear ()
- void incCountValue (double value=1.0)
- double getCountValue () const
- ModelElement ∗ getParent () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _initBetweenReplications ()

Additional Inherited Members

7.23.1 Detailed Description

The Counter element is used to count events, and its internal count value is added by a configurable amount, usually incremented by one.

Definition at line 25 of file Counter.h.

7.23.2 Constructor & Destructor Documentation

7.23.2.1 Counter()

```
Counter::Counter (
    Model * model,
    std::string name = "",
    ModelElement * parent = nullptr )
```

Definition at line 20 of file Counter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.23.2.2 ~Counter() `virtual Counter::~Counter ( ) [virtual], [default]`

7.23.3 Member Function Documentation

7.23.3.1 _check() `bool Counter::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 75 of file Counter.cpp.

7.23.3.2 _initBetweenReplications() `void Counter::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 80 of file Counter.cpp.

7.23.3.3 _loadInstance() `bool Counter::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 67 of file Counter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.23.3.4 _saveInstance() `std::map< std::string, std::string > * Counter::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 71 of file Counter.cpp.

Here is the call graph for this function:

7.23.3.5 clear() `void Counter::clear ( )`

Definition at line 35 of file Counter.cpp.

Here is the caller graph for this function:

7.23.3.6 getCountValue() `double Counter::getCountValue ( ) const`

Definition at line 43 of file Counter.cpp.

Here is the caller graph for this function:

7.23.3.7 getParent() `ModelElement * Counter::getParent ( ) const`

Definition at line 47 of file Counter.cpp.

Here is the caller graph for this function:

7.23.3.8 GetPluginInformation() `PluginInformation * Counter::GetPluginInformation ( ) [static]`

Definition at line 51 of file Counter.cpp.

Here is the call graph for this function:

7.23.3.9 incCountValue() `void Counter::incCountValue ( double value = 1.0 )`

Definition at line 39 of file Counter.cpp.

Here is the caller graph for this function:

7.23.3.10 LoadInstance()

```
ModelElement * Counter::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 57 of file Counter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.23.3.11 show() `std::string Counter::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 30 of file Counter.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Counter.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Counter.cpp

7.24 Create Class Reference

```
#include <Create.h>
```

Inheritance diagram for Create:

Collaboration diagram for Create:

Public Member Functions

- Create (Model ∗model, std::string name="")
- virtual ∼Create ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.24.1 Detailed Description

Create is the most basic component to include the first entities into the model, and therefore is a source component (derived from SourceModelComponent) Create module DESCRIPTION This module is intended as the starting point for entities in a simulation model. Entities are created using a schedule or based on a time between arrivals. Entities then leave the module to begin processing through the system. The entity type is specified in this module. TYPICAL USES The start of a part's production in a manufacturing line A document's arrival (for example, order, check, application) into a business process A customer's arrival at a service process (for example, retail store, restaurant, information desk) PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Entity Type Name of the entity type to be generated. Type Type of arrival stream to be generated. Types include Random (uses an exponential distribution, user specifies mean), Schedule (uses an exponential distribution, mean determined from the specified Schedule module), Constant (user specifies constant value; for example, 100), or Expression (drop-down list of various distributions). Value Determines the mean of the exponential distribution (if Random is used) or the constant value (if Constant is used) for the time between arrivals. Applies only when Type is Random or Constant. Schedule Name Identifies the name of the schedule to be used. The schedule defines the arrival pattern for entities arriving to the system. Applies only when Type is Schedule. Expression Any distribution or value specifying the time between arrivals. Applies only when Type is Expression. Units Time units used for interarrival and first creation times. Does not apply when Type is Schedule. Entities per Arrival Number of entities that will enter the system at a given time with each arrival. Max Arrivals Maximum number of entities that this module will generate. When this value is reached, the creation of new entities by this module ceases. First Creation Starting time for the first entity to arrive into the system. Does not apply when Type is Schedule.

Definition at line 67 of file Create.h.

7.24.2 Constructor & Destructor Documentation

7.24.2.1 Create()

```
Create::Create (
    Model * model,
    std::string name = "" )
```

Definition at line 21 of file Create.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.24.2.2 ~Create() `virtual Create::~Create ( ) [virtual], [default]`

7.24.3 Member Function Documentation

7.24.3.1 _check() `bool Create::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from SourceModelComponent.

Definition at line 101 of file Create.cpp.

Here is the call graph for this function:

7.24.3.2 _createInternalElements() `void Create::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 106 of file Create.cpp.

Here is the call graph for this function:

7.24.3.3 _execute() `void Create::_execute ( Entity * entity ) [protected], [virtual]`

Implements ModelComponent.

Definition at line 41 of file Create.cpp.

7.24.3.4 _initBetweenReplications() `void Create::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from SourceModelComponent.

Definition at line 92 of file Create.cpp.

Here is the call graph for this function:

7.24.3.5 _loadInstance() `bool Create::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from SourceModelComponent.

Definition at line 88 of file Create.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.24.3.6 _saveInstance() `std::map< std::string, std::string > * Create::_saveInstance ( ) [protected], [virtual]`

Reimplemented from SourceModelComponent.

Definition at line 96 of file Create.cpp.

Here is the call graph for this function:

7.24.3.7 GetPluginInformation() `PluginInformation * Create::GetPluginInformation ( ) [static]`

Definition at line 69 of file Create.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.24.3.8 LoadInstance() `ModelComponent * Create::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 78 of file Create.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.24.3.9 show() `std::string Create::show ( ) [virtual]`

Reimplemented from SourceModelComponent.

Definition at line 37 of file Create.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Create.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Create.cpp

7.25 Decide Class Reference

```
#include <Decide.h>
```

Inheritance diagram for Decide:

Collaboration diagram for Decide:

Public Member Functions

- Decide (Model ∗model, std::string name="")
- virtual ∼Decide ()=default
- List< std::string > ∗ getConditions () const
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.25.1 Detailed Description

Decide module DESCRIPTION This module allows for decision-making processes in the system. It includes options to make decisions based on one or more conditions (for example, if entity type is Gold Card) or based on one or more probabilities (for example, 75%, true; 25%, false). Conditions can be based on attribute values (for example, Priority), variable values (for example, Number Denied), the entity type, or an expression (for example, NQ(ProcessA.Queue)). There are two exit points out of the Decide module when its specified type is either 2-way by Chance or 2-way by Condition. There is one exit point for "true" entities and one for "false" entities. When the N-way by Chance or by Condition type is specified, multiple exit points are shown for each condition or probability and a single "else" exit. The number of entities that exit from each type (true/false) is displayed for 2-way by Chance or by Condition modules only. TYPICAL USES Dispatching a faulty part for rework Branching accepted vs. rejected checks Sending priority customers to a dedicated process Prompt Description Name Unique module identifier displayed on the module shape. Type Indicates whether the decision is based on a condition (if X>Y) or by chance/percentage (for example, 60%, yes; 40%, no). The type can be specified as either 2-way or N-way. 2-way allows for one condition or probability (plus the "false" exit). N-way allows for any number of conditions or probabilities to be specified as well as an "else" exit. Conditions Defines one or more conditions used to direct entities to different modules. Applies only when Type is N-way by Condition. Percentages Defines one or more percentages used to direct entities to different modules. Applies only when Type is N-way by Chance. Percent True Value that will be checked to determine the percentage of entities sent out a given True exit. If Types of conditions that are available for evaluation: Variable, Variable Array (1D), Variable Array (2D), Attribute, Entity Type, Expression. Named Specifies the name of the variable, attribute, or entity type that will be evaluated when an entity enters the module. Does not apply when Type is Expression. Is Evaluator for the condition. Applies only to Attribute and Variable conditions. Row Specifies the row index for a variable array. Applies only when Type is N-way by Condition or 2-way by Condition and Variable is Array 1-D or Array 2-D. Column Specifies the column index for a variable array. Applies only when Type is N-way by Condition or 2-way by Condition and Variable is Array 1-D or Array 2-D. Value Expression that will be either compared to an attribute or variable or that will be evaluated as a single expression to determine if it is true or false. Does not apply to Entity Type condition. If Type is Expression, this value must also include the evaluator (for example, Color<>Red).

Definition at line 74 of file Decide.h.

7.25.2 Constructor & Destructor Documentation

7.25.2.1 Decide()

```
Decide::Decide (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Decide.cpp.

Here is the caller graph for this function:

7.25.2.2 ~Decide() `virtual Decide::~Decide ( ) [virtual], [default]`

7.25.3 Member Function Documentation

7.25.3.1 _check() `bool Decide::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 80 of file Decide.cpp.

Here is the call graph for this function:

7.25.3.2 _createInternalElements() `void Decide::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 90 of file Decide.cpp.

Here is the call graph for this function:

7.25.3.3 _execute()

```
void Decide::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 29 of file Decide.cpp.

Here is the call graph for this function:

7.25.3.4 _initBetweenReplications()

```
void Decide::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 51 of file Decide.cpp.

Here is the call graph for this function:

7.25.3.5 _loadInstance()

```
bool Decide::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 59 of file Decide.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.25.3.6 _saveInstance()

```
std::map< std::string, std::string > * Decide::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 70 of file Decide.cpp.

Here is the call graph for this function:

7.25.3.7 getConditions() `List< std::string > * Decide::getConditions ( ) const`

Definition at line 21 of file Decide.cpp.

Here is the caller graph for this function:

7.25.3.8 GetPluginInformation() `PluginInformation * Decide::GetPluginInformation ( ) [static]`

Definition at line 105 of file Decide.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.25.3.9 LoadInstance()

```
ModelComponent * Decide::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 113 of file Decide.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.25.3.10 show()

```
std::string Decide::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 25 of file Decide.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Decide.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Decide.cpp

7.26 ModelComponent::DEFAULT_VALUES Struct Reference

```
#include <ModelComponent.h>
```

Collaboration diagram for ModelComponent::DEFAULT_VALUES:

Public Attributes

- unsigned int nextSize = 1
- unsigned int nextInputNumber = 0
- std::string description = ""

7.26.1 Detailed Description

Definition at line 58 of file ModelComponent.h.

7.26.2 Member Data Documentation

7.26.2.1 description `std::string ModelComponent::DEFAULT_VALUES::description = ""`

Definition at line 61 of file ModelComponent.h.

7.26.2.2 nextInputNumber `unsigned int ModelComponent::DEFAULT_VALUES::nextInputNumber = 0`

Definition at line 60 of file ModelComponent.h.

7.26.2.3 nextSize `unsigned int ModelComponent::DEFAULT_VALUES::nextSize = 1`

Definition at line 59 of file ModelComponent.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelComponent.h

7.27 SourceModelComponent::DEFAULT_VALUES Struct Reference

`#include <SourceModelComponent.h>`

Collaboration diagram for SourceModelComponent::DEFAULT_VALUES:

Public Attributes

- const double firstCreation = 0.0
- const unsigned int entitiesPerCreation = 1
- const std::string maxCreationsExpression = std::to_string(std::numeric_limits<unsigned int>::max())
- const std::string timeBetweenCreationsExpression = "EXPO(1.0)"
- const Util::TimeUnit timeBetweenCreationsTimeUnit = Util::TimeUnit::second

7.27.1 Detailed Description

Definition at line 57 of file SourceModelComponent.h.

7.27.2 Member Data Documentation

7.27.2.1 entitiesPerCreation

`const unsigned int SourceModelComponent::DEFAULT_VALUES::entitiesPerCreation = 1`

Definition at line 59 of file SourceModelComponent.h.

7.27.2.2 firstCreation

`const double SourceModelComponent::DEFAULT_VALUES::firstCreation = 0.0`

Definition at line 58 of file SourceModelComponent.h.

7.27.2.3 maxCreationsExpression

`const std::string SourceModelComponent::DEFAULT_VALUES::maxCreationsExpression = std::to_string(std::numeric_limits<unsigned int>::max())`

Definition at line 60 of file SourceModelComponent.h.

7.27.2.4 timeBetweenCreationsExpression

`const std::string SourceModelComponent::DEFAULT_VALUES::timeBetweenCreationsExpression = "EXPO(1.0)"`

Definition at line 61 of file SourceModelComponent.h.

7.27.2.5 timeBetweenCreationsTimeUnit

`const Util::TimeUnit SourceModelComponent::DEFAULT_VALUES::timeBetweenCreationsTimeUnit = Util::TimeUnit::second`

Definition at line 62 of file SourceModelComponent.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SourceModelComponent.h

7.28 WriteExpression::DEFAULT_VALUES Struct Reference

```
#include <Write.h>
```

Collaboration diagram for WriteExpression::DEFAULT_VALUES:

Public Attributes

- const std::string text = ""
- const bool isExpression = true
- const bool newline = false

7.28.1 Detailed Description

Definition at line 56 of file Write.h.

7.28.2 Member Data Documentation

7.28.2.1 isExpression `const bool WriteExpression::DEFAULT_VALUES::isExpression = true`

Definition at line 58 of file Write.h.

7.28.2.2 newline `const bool WriteExpression::DEFAULT_VALUES::newline = false`

Definition at line 59 of file Write.h.

7.28.2.3 text `const std::string WriteExpression::DEFAULT_VALUES::text = ""`

Definition at line 57 of file Write.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.29 WriteInExpression::DEFAULT_VALUES Struct Reference

```
#include <Write.h>
```

Collaboration diagram for WriteInExpression::DEFAULT_VALUES:

Public Attributes

- const std::string text = ""
- const bool isExpression = true
- const bool newline = true

7.29.1 Detailed Description

Definition at line 70 of file Write.h.

7.29.2 Member Data Documentation

7.29.2.1 isExpression `const bool WriteInExpression::DEFAULT_VALUES::isExpression = true`

Definition at line 72 of file Write.h.

7.29.2.2 newline `const bool WriteInExpression::DEFAULT_VALUES::newline = true`

Definition at line 73 of file Write.h.

7.29.2.3 text `const std::string WriteInExpression::DEFAULT_VALUES::text = ""`

Definition at line 71 of file Write.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.30 WriteInText::DEFAULT_VALUES Struct Reference

`#include <Write.h>`

Collaboration diagram for WriteInText::DEFAULT_VALUES:

Public Attributes

- const std::string text = ""
- const bool isExpression = false
- const bool newline = true

7.30.1 Detailed Description

Definition at line 43 of file Write.h.

7.30.2 Member Data Documentation

7.30.2.1 isExpression `const bool WritelnText::DEFAULT_VALUES::isExpression = false`

Definition at line 45 of file Write.h.

7.30.2.2 newline `const bool WritelnText::DEFAULT_VALUES::newline = true`

Definition at line 46 of file Write.h.

7.30.2.3 text `const std::string WritelnText::DEFAULT_VALUES::text = ""`

Definition at line 44 of file Write.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.31 WriteText::DEFAULT_VALUES Struct Reference

```
#include <Write.h>
```

Collaboration diagram for WriteText::DEFAULT_VALUES:

Public Attributes

- const std::string text = ""
- const bool isExpression = false
- const bool newline = false

7.31.1 Detailed Description

Definition at line 27 of file Write.h.

7.31.2 Member Data Documentation

7.31.2.1 isExpression `const bool WriteText::DEFAULT_VALUES::isExpression = false`

Definition at line 29 of file Write.h.

7.31.2.2 newline `const bool WriteText::DEFAULT_VALUES::newline = false`

Definition at line 30 of file Write.h.

7.31.2.3 text `const std::string WriteText::DEFAULT_VALUES::text = ""`

Definition at line 28 of file Write.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.32 SamplerDefaultImpl1::DefaultImpl1RNG_Parameters Struct Reference

```
#include <SamplerDefaultImpl1.h>
```

Inheritance diagram for SamplerDefaultImpl1::DefaultImpl1RNG_Parameters:

Collaboration diagram for SamplerDefaultImpl1::DefaultImpl1RNG_Parameters:

Public Member Functions

- ~DefaultImpl1RNG_Parameters ()=default

Public Attributes

- uint32_t seed = 16021974
- uint32_t a = 279470273u
- uint32_t m = 0xfffffffb
- uint32_t c = 0

7.32.1 Detailed Description

Definition at line 23 of file SamplerDefaultImpl1.h.

7.32.2 Constructor & Destructor Documentation

7.32.2.1 ~DefaultImpl1RNG_Parameters() `SamplerDefaultImpl1::DefaultImpl1RNG_Parameters::~DefaultImpl1RNG_Parameters ( ) [default]`

7.32.3 Member Data Documentation

7.32.3.1 a `uint32_t SamplerDefaultImpl1::DefaultImpl1RNG_Parameters::a = 279470273u`

Definition at line 25 of file SamplerDefaultImpl1.h.

7.32.3.2 c `uint32_t SamplerDefaultImpl1::DefaultImpl1RNG_Parameters::c = 0`

Definition at line 27 of file SamplerDefaultImpl1.h.

7.32.3.3 m `uint32_t SamplerDefaultImpl1::DefaultImpl1RNG_Parameters::m = 0xfffffffb`

Definition at line 26 of file SamplerDefaultImpl1.h.

7.32.3.4 seed `uint32_t SamplerDefaultImpl1::DefaultImpl1RNG_Parameters::seed = 16021974`

Definition at line 24 of file SamplerDefaultImpl1.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.h

7.33 Delay Class Reference

```
#include <Delay.h>
```

Inheritance diagram for Delay:

Collaboration diagram for Delay:

Public Member Functions

- Delay (Model ∗model, std::string name="")
- virtual ∼Delay ()=default
- void setDelayExpression (std::string _delayExpression)
- std::string delayExpression () const
- void setDelay (double delay)
- double delay () const
- void setDelayTimeUnit (Util::TimeUnit _delayTimeUnit)
- Util::TimeUnit delayTimeUnit () const
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.33.1 Detailed Description

Delay module DESCRIPTION The Delay module delays an entity by a specified amount of time. When an entity arrives at a Delay module, the time delay expression is evaluated and the entity remains in the module for the resulting time period. The time is then allocated to the entity's value-added, non-value added, transfer, wait, or other time. Associated costs are calculated and allocated as well. TYPICAL USES Processing a check at a bank Performing a setup on a machine Transferring a document to another department PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Allocation Type of category to which the entity's incurred delay time and cost will be added. Delay Time Determines the value of the delay for the entity. Units Time units used for the delay time.

Definition at line 41 of file Delay.h.

7.33.2 Constructor & Destructor Documentation

7.33.2.1 Delay()

```
Delay::Delay (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Delay.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.33.2.2 ~Delay()

```
virtual Delay::~Delay ( ) [virtual], [default]
```

7.33.3 Member Function Documentation

7.33.3.1 _check()

```
bool Delay::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 107 of file Delay.cpp.

Here is the call graph for this function:

7.33.3.2 _createInternalElements()

```
void Delay::_createInternalElements ( ) [protected], [virtual]
```

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 122 of file Delay.cpp.

Here is the call graph for this function:

7.33.3.3 _execute()

```
void Delay::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 71 of file Delay.cpp.

7.33.3.4 _initBetweenReplications()

```
void Delay::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 97 of file Delay.cpp.

7.33.3.5 _loadInstance()

```
bool Delay::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 88 of file Delay.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.33.3.6 _saveInstance() `std::map< std::string, std::string > * Delay::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 100 of file Delay.cpp.

Here is the call graph for this function:

7.33.3.7 delay() `double Delay::delay ( ) const`

Definition at line 35 of file Delay.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.33.3.8 delayExpression() `std::string Delay::delayExpression ( ) const`

Definition at line 59 of file Delay.cpp.

Here is the caller graph for this function:

7.33.3.9 delayTimeUnit() `Util::TimeUnit Delay::delayTimeUnit ( ) const`

Definition at line 67 of file Delay.cpp.

Here is the caller graph for this function:

7.33.3.10 GetPluginInformation() `PluginInformation * Delay::GetPluginInformation ( ) [static]`

Definition at line 140 of file Delay.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.33.3.11 LoadInstance()

```
ModelComponent * Delay::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 45 of file Delay.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.33.3.12 setDelay()

```
void Delay::setDelay (
    double delay )
```

Definition at line 31 of file Delay.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.33.3.13 setDelayExpression()

```
void Delay::setDelayExpression (
    std::string _delayExpression )
```

Definition at line 55 of file Delay.cpp.

Here is the caller graph for this function:

7.33.3.14 setDelayTimeUnit()

```
void Delay::setDelayTimeUnit (
    Util::TimeUnit _delayTimeUnit )
```

Definition at line 63 of file Delay.cpp.

Here is the caller graph for this function:

7.33.3.15 show() `std::string Delay::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 39 of file Delay.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Delay.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Delay.cpp

7.34 Ui::DialogAbout Class Reference

```
#include <ui_dialogabout.h>
```

Inheritance diagram for Ui::DialogAbout:

Collaboration diagram for Ui::DialogAbout:

Additional Inherited Members

7.34.1 Detailed Description

Definition at line 135 of file ui_dialogabout.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/ui_dialogabout.h

7.35 Ui::DialogModelInformation Class Reference

```
#include <ui_dialogmodelinformation.h>
```

Inheritance diagram for Ui::DialogModelInformation:

Collaboration diagram for Ui::DialogModelInformation:

Additional Inherited Members

7.35.1 Detailed Description

Definition at line 167 of file ui_dialogmodelinformation.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/ui_dialogmodelinformation.h

7.36 Dispose Class Reference

```
#include <Dispose.h>
```

Inheritance diagram for Dispose:

Collaboration diagram for Dispose:

Public Member Functions

- Dispose (Model *model, std::string name="")
- virtual ~Dispose ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelComponent * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.36.1 Detailed Description

Dispose module DESCRIPTION This module is intended as the ending point for entities in a simulation model. Entity statistics may be recorded before the entity is disposed of. TYPICAL USES Parts leaving the modeled facility The termination of a business process Customers departing from the store Prompt Description Name Unique module identifier displayed on the module shape. Record Entity Statistics Determines whether or not the incoming entity's statistics will be recorded. Statistics include value-added time, non-value-added time, wait time, transfer time, other time, total time, value-added cost, non-value-added cost, wait cost, transfer cost, other cost, and total cost.

Definition at line 38 of file Dispose.h.

7.36.2 Constructor & Destructor Documentation

7.36.2.1 Dispose()

```
Dispose::Dispose (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Dispose.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.36.2.2 ~Dispose()

```
virtual Dispose::~Dispose ( ) [virtual], [default]
```

7.36.3 Member Function Documentation

7.36.3.1 _check()

```
bool Dispose::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from SinkModelComponent.

Definition at line 52 of file Dispose.cpp.

7.36.3.2 _createInternalElements()

```
void Dispose::_createInternalElements ( ) [protected], [virtual]
```

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 57 of file Dispose.cpp.

Here is the call graph for this function:

7.36.3.3 _execute()

```
void Dispose::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 27 of file Dispose.cpp.

Here is the call graph for this function:

7.36.3.4 _initBetweenReplications() `void Dispose::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from SinkModelComponent.

Definition at line 42 of file Dispose.cpp.

Here is the call graph for this function:

7.36.3.5 _loadInstance() `bool Dispose::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from SinkModelComponent.

Definition at line 38 of file Dispose.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.36.3.6 _saveInstance() `std::map< std::string, std::string > * Dispose::_saveInstance ( ) [protected], [virtual]`

Reimplemented from SinkModelComponent.

Definition at line 46 of file Dispose.cpp.

Here is the call graph for this function:

7.36.3.7 GetPluginInformation() `PluginInformation * Dispose::GetPluginInformation ( ) [static]`

Definition at line 76 of file Dispose.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.36.3.8 LoadInstance()

```
ModelComponent * Dispose::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 82 of file Dispose.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.36.3.9 show()

```
std::string Dispose::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 23 of file Dispose.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Dispose.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Dispose.cpp

7.37 DropOff Class Reference

```
#include <DropOff.h>
```

Inheritance diagram for DropOff:

Collaboration diagram for DropOff:

Public Member Functions

- DropOff (Model ∗model, std::string name="")
- virtual ∼DropOff ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.37.1 Detailed Description

Dropoff module DESCRIPTION The Dropoff module removes a specified number of entities from the entity's group and sends them to another module, as specified by a graphical connection. Group user-defined attribute value and internal attributes may be given to the dropped-off entities based on a specified rule. TYPICAL USES Loading shelves with product Separating a form for use in various departments PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Quantity Number of entities that will be dropped off from an incoming representative grouped entity. Starting Rank Starting rank of the entities to be dropped off, based on the entities in the group. Member Attributes Method of determining how to assign the representative entity attribute values (other than costs/times) to the dropped-off original entities. Attribute Name Name of representative entity attribute(s) assigned to droppedoff original entities of the group

Definition at line 41 of file DropOff.h.

7.37.2 Constructor & Destructor Documentation

7.37.2.1 DropOff()

```
DropOff::DropOff (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file DropOff.cpp.

Here is the caller graph for this function:

7.37.2.2 ~DropOff()

```
virtual DropOff::~DropOff ( ) [virtual], [default]
```

7.37.3 Member Function Documentation

7.37.3.1 _check()

```
bool DropOff::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file DropOff.cpp.

7.37.3.2 _execute()

```
void DropOff::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file DropOff.cpp.

Here is the call graph for this function:

7.37.3.3 _initBetweenReplications()

```
void DropOff::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 47 of file DropOff.cpp.

7.37.3.4 _loadInstance()

```
bool DropOff::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 39 of file DropOff.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.37.3.5 _saveInstance() `std::map< std::string, std::string > * DropOff::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 50 of file DropOff.cpp.

Here is the call graph for this function:

7.37.3.6 GetPluginInformation() `PluginInformation * DropOff::GetPluginInformation ( ) [static]`

Definition at line 63 of file DropOff.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.37.3.7 LoadInstance()

```
ModelComponent * DropOff::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 24 of file DropOff.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.37.3.8 show()

```
std::string DropOff::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 20 of file DropOff.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/DropOff.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/DropOff.cpp

7.38 Dummy Class Reference

```
#include <Dummy.h>
```

Inheritance diagram for Dummy:

Collaboration diagram for Dummy:

Public Member Functions

- Dummy (Model ∗model, std::string name="")
- virtual ∼Dummy ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.38.1 Detailed Description

This component ...

Definition at line 22 of file Dummy.h.

7.38.2 Constructor & Destructor Documentation

7.38.2.1 Dummy()

```
Dummy::Dummy (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Dummy.cpp.

Here is the caller graph for this function:

7.38.2.2 ~Dummy()

```
virtual Dummy::~Dummy ( ) [virtual], [default]
```

7.38.3 Member Function Documentation

7.38.3.1 _check()

```
bool Dummy::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file Dummy.cpp.

7.38.3.2 _execute()

```
void Dummy::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file Dummy.cpp.

Here is the call graph for this function:

7.38.3.3 _initBetweenReplications() `void Dummy::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 47 of file Dummy.cpp.

7.38.3.4 _loadInstance() `bool Dummy::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 39 of file Dummy.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.38.3.5 _saveInstance() `std::map< std::string, std::string > * Dummy::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 50 of file Dummy.cpp.

Here is the call graph for this function:

7.38.3.6 GetPluginInformation() `PluginInformation * Dummy::GetPluginInformation ( ) [static]`

Definition at line 63 of file Dummy.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.38.3.7 LoadInstance()

```
ModelComponent * Dummy::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 24 of file Dummy.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.38.3.8 show()

```
std::string Dummy::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 20 of file Dummy.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Dummy.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Dummy.cpp

7.39 ElementManager Class Reference

```
#include <ElementManager.h>
```

Collaboration diagram for ElementManager:

Public Member Functions

- ElementManager (Model ∗model)
- virtual ∼ElementManager ()=default
- bool insert (ModelElement ∗anElement)
- void remove (ModelElement ∗anElement)

  Deprected.
- bool insert (std::string elementTypename, ModelElement ∗anElement)

  Deprected.
- void remove (std::string elementTypename, ModelElement ∗anElement)
- bool check (std::string elementTypename, ModelElement ∗anElement, std::string expressionName, std::string ∗errorMessage)
- bool check (std::string elementTypename, std::string elementName, std::string expressionName, bool mandatory, std::string ∗errorMessage)
- void clear ()
- ModelElement ∗ getElement (std::string elementTypename, Util::identification id)
- ModelElement ∗ getElement (std::string elementTypename, std::string name)
- unsigned int getNumberOfElements (std::string elementTypename)
- unsigned int getNumberOfElements ()
- int getRankOf (std::string elementTypename, std::string name)

  returns the position (1st position=0) of the element if found, or negative value if not found
- std::list< std::string > ∗ getElementClassnames () const
- List< ModelElement ∗ > ∗ getElementList (std::string elementTypename) const
- void show ()
- Model ∗ getParentModel () const
- bool hasChanged () const
- void setHasChanged (bool _hasChanged)

7.39.1 Detailed Description

The ElementManager is responsible for inserting and removing elements (ModelElement) used by components, in a consistent way. TO FIX: No direct access for insertion or deletion should be allow

Definition at line 30 of file ElementManager.h.

7.39.2 Constructor & Destructor Documentation

7.39.2.1 ElementManager()

```
ElementManager::ElementManager (
    Model * model )
```

Elements are organized as a map from a string (key), the type of an element, and a list of elements of that type

Definition at line 19 of file ElementManager.cpp.

7.39.2.2 ~ElementManager()

```
virtual ElementManager::~ElementManager ( ) [virtual], [default]
```

7.39.3 Member Function Documentation

7.39.3.1 check() [1/2]

```
bool ElementManager::check (
    std::string elementTypename,
    ModelElement * anElement,
    std::string expressionName,
    std::string * errorMessage )
```

Definition at line 70 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.2 check() [2/2]

```
bool ElementManager::check (
    std::string elementTypename,
    std::string elementName,
    std::string expressionName,
    bool mandatory,
    std::string * errorMessage )
```

Definition at line 58 of file ElementManager.cpp.

Here is the call graph for this function:

7.39.3.3 clear()

```
void ElementManager::clear ( )
```

Definition at line 81 of file ElementManager.cpp.

Here is the caller graph for this function:

7.39.3.4 getElement() [1/2]

```
ModelElement * ElementManager::getElement (
    std::string elementTypename,
    std::string name )
```

Definition at line 191 of file ElementManager.cpp.

Here is the call graph for this function:

7.39.3.5 getElement() [2/2]

```
ModelElement * ElementManager::getElement (
    std::string elementTypename,
    Util::identification id )
```

Definition at line 160 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.6 getElementClassnames() `std::list< std::string > * ElementManager::getElementClassnames ( ) const`

Definition at line 183 of file ElementManager.cpp.

Here is the caller graph for this function:

7.39.3.7 getElementList() `List< ModelElement * > * ElementManager::getElementList ( std::string elementTypename ) const`

Definition at line 145 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.8 getNumberOfElements() [1/2] `unsigned int ElementManager::getNumberOfElements ( )`

Definition at line 91 of file ElementManager.cpp.

7.39.3.9 getNumberOfElements() [2/2]

```
unsigned int ElementManager::getNumberOfElements (
    std::string elementTypename )
```

Definition at line 86 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.10 getParentModel()

```
Model * ElementManager::getParentModel ( ) const
```

Definition at line 122 of file ElementManager.cpp.

7.39.3.11 getRankOf()

```
int ElementManager::getRankOf (
    std::string elementTypename,
    std::string name )
```

returns the position (1st position=0) of the element if found, or negative value if not found

Definition at line 170 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.12 hasChanged() `bool ElementManager::hasChanged ( ) const`

Definition at line 126 of file ElementManager.cpp.

Here is the caller graph for this function:

7.39.3.13 insert() [1/2] `bool ElementManager::insert ( ModelElement * anElement )`

Definition at line 25 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.14 insert() [2/2]

```
bool ElementManager::insert (
    std::string elementTypename,
    ModelElement * anElement )
```

Deprected.

Definition at line 30 of file ElementManager.cpp.

Here is the call graph for this function:

7.39.3.15 remove() [1/2]

```
void ElementManager::remove (
    ModelElement * anElement )
```

Deprected.

Definition at line 43 of file ElementManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.39.3.16 remove() [2/2]

```
void ElementManager::remove (
    std::string elementTypename,
    ModelElement * anElement )
```

Definition at line 52 of file ElementManager.cpp.

Here is the call graph for this function:

7.39.3.17 setHasChanged() `void ElementManager::setHasChanged ( bool _hasChanged )`

Definition at line 141 of file ElementManager.cpp.

7.39.3.18 show() `void ElementManager::show ( )`

Definition at line 99 of file ElementManager.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ElementManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ElementManager.cpp

7.40 ElementManager_if Class Reference

```
#include <ElementManager_if.h>
```

Collaboration diagram for ElementManager_if:

Public Member Functions

- ElementManager_if ()
- ElementManager_if (const ElementManager_if &orig)
- virtual ~ElementManager_if ()

7.40.1 Detailed Description

Definition at line 17 of file ElementManager_if.h.

7.40.2 Constructor & Destructor Documentation

7.40.2.1 ElementManager_if() [1/2] `ElementManager_if::ElementManager_if ( )`

7.40.2.2 ElementManager_if() [2/2] `ElementManager_if::ElementManager_if ( const ElementManager_if & orig )`

7.40.2.3 ~ElementManager_if() `virtual ElementManager_if::~ElementManager_if ( ) [virtual]`

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ElementManager_if.h

7.41 Enter Class Reference

```
#include <Enter.h>
```

Inheritance diagram for Enter:

Collaboration diagram for Enter:

Public Member Functions

- Enter (Model ∗model, std::string name="")
- virtual ∼Enter ()=default
- virtual std::string show ()
- void setStation (Station ∗_station)
- Station ∗ getStation () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.41.1 Detailed Description

Enter module DESCRIPTION The Enter module defines a station (or a set of stations) corresponding to a physical or logical location where processing occurs. When an entity arrives at an Enter module, an unloading delay may occur and any transfer device used to transfer the entity to the Enter module's station may be released. The station (or each station within the defined set) has a matching Activity Area that is used to report all times and costs accrued by the entities in this station. This Activity Area's name is the same as the station. If a parent Activity Area is defined, then it also accrues any times and costs by the entities in this station. TYPICAL USES The start of a part's production in a series of parallel processes where the part's forklift needs to be released The start of a document's processing after the document has been created where the mail clerk resource needs to be released PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart Station Type Type of station, either a single Station or station Set. Station Name Name of the individual station. A given station can only exist once within a model. Parent Activity Area Name of the Activity Area's parent. Associated Intersection Name of the intersection associated with this station in a guided transporter network. Report Statistics Specifies whether or not statistics will automatically be collected and stored in the report database for this station and its corresponding activity area. Set Name Name of the station set. A given station set can only exist once within a model. Save Attribute Specifies the attribute to be used to store the index into the station set for an entity entering this module. Set Members This repeat group permits you to define the individual stations that are to be members of the specified station set. A station set must have at least one member station. Active when Station Type is Set. Station Name This field indicates the name of a station that is to be a member of this station set. A given station can only exist within a model once. Therefore, an individual station can only be the member of one station set, and that individual station may not be the name of a station in another module. Parent Activity Area Name of the Activity Area's parent for the station set member. Associated Intersection Name of the intersection associated with this station set in a guided transporter network. Report Statistics Specifies whether or not statistics will automatically be collected and stored in the report database for this station set member and its corresponding activity area. Allocation Type of category to which the entity's incurred delay time and cost will be added. Delay This field defines the delay that will be experienced by entities immediately upon arrival at the station. Units Time units used for the delay time. Transfer In If a resource, transporter, or conveyor was used to transfer the entity to this station, this can be used to release, free, or exit the device. If Release Resource is selected, the specified resource is released. If Free Transporter is selected, the specified transporter is freed. If Exit Conveyor is selected, the specified conveyor is exited. Transporter Name Name of the transporter to be freed upon arrival to the station. Active when Transfer Name is Free Transporter. Unit Number Unit number of the transporter if the transporter is multicapacity. Conveyor Name Name of the conveyor to exit upon arrival to the station. Resource Type Type of allocation, either single Resource or resource Set. Resource Name Name of the resource to release. Active when Transfer Name is Release Resource. Set Name Name of the resource set from which the resource is to be released. Release Rule Determines which member of the set is to be released, either the Last Member Seized, First Member Seized, or Specific Member. Set Index Index into the set that determines which member of the set is to be released. Attribute Name Name of the attribute that determines the instance number of the resource to release. Expression Expression value that determines the instance number of the resource to release.

Definition at line 99 of file Enter.h.

7.41.2 Constructor & Destructor Documentation

7.41.2.1 Enter()

```
Enter::Enter (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Enter.cpp.

Here is the caller graph for this function:

7.41.2.2 ~Enter() `virtual Enter::~Enter ( ) [virtual], [default]`

7.41.3 Member Function Documentation

7.41.3.1 _check() `bool Enter::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 71 of file Enter.cpp.

Here is the call graph for this function:

7.41.3.2 _createInternalElements() `void Enter::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 84 of file Enter.cpp.

Here is the call graph for this function:

7.41.3.3 _execute()

```
void Enter::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 44 of file Enter.cpp.

Here is the call graph for this function:

7.41.3.4 _initBetweenReplications()

```
void Enter::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 61 of file Enter.cpp.

Here is the call graph for this function:

7.41.3.5 _loadInstance() `bool Enter::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Enter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.41.3.6 _saveInstance() `std::map< std::string, std::string > * Enter::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 65 of file Enter.cpp.

Here is the call graph for this function:

7.41.3.7 GetPluginInformation() `PluginInformation * Enter::GetPluginInformation ( ) [static]`

Definition at line 77 of file Enter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.41.3.8 getStation() `Station * Enter::getStation ( ) const`

Definition at line 40 of file Enter.cpp.

7.41.3.9 LoadInstance() `ModelComponent * Enter::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 25 of file Enter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.41.3.10 setStation() `void Enter::setStation ( Station * _station )`

Definition at line 35 of file Enter.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.41.3.11 show() `std::string Enter::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file Enter.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Enter.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Enter.cpp

7.42 Entity Class Reference

```
#include <Entity.h>
```

Inheritance diagram for Entity:

Collaboration diagram for Entity:

Public Member Functions

- virtual std::string show ()
- void setEntityTypeName (std::string entityTypeName) throw ()
- std::string getEntityTypeName () const
- void setEntityType (EntityType *entityType)
- EntityType * getEntityType () const
- double getAttributeValue (std::string attributeName)
- double getAttributeValue (std::string index, std::string attributeName)
- double getAttributeValue (Util::identification attributeID)
- double getAttributeValue (std::string index, Util::identification attributeID)
- void setAttributeValue (std::string attributeName, double value)
- void setAttributeValue (std::string index, std::string attributeName, double value)
- void setAttributeValue (Util::identification attributeID, double value)
- void setAttributeValue (std::string index, Util::identification attributeID, double value)
- Util::identification entityNumber () const

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)

Friends

- class Model

Additional Inherited Members

7.42.1 Detailed Description

Entity module DESCRIPTION This data module defines the various entity types and their initial picture values in a simulation. Initial costing information and holding costs are also defined for the entity. TYPICAL USES Items being produced or assembled (parts, pallets) Documents (forms, e-mails, faxes, reports) People moving through a process (customers, callers) PROMPTS Prompt Description Name The unique name of the attribute being defined. Rows Number of rows in a one- or two-dimensional attribute. Columns Number of columns in a two-dimensional attribute. Data Type The data type of the values stored in the attribute. Valid types are Real and String. The default type is Real. Initial Values Lists the initial value or values of the attribute. You can assign new values to the attribute by using the Assign module. Initial Value Entity attribute value when entity is created and enters the system. Prompt Description Entity Type The name of the entity type being defined. This name must be unique. Initial Picture Graphical representation of the entity at the start of the simulation. This value can be changed during the simulation using the Assign module. Holding Cost/Hour Hourly cost of processing the entity through the system. This cost is incurred when the entity is anywhere in the system. Initial VA Cost Initial cost value that will be assigned to the value-added cost attribute of the entity. This attribute accrues the costs incurred when an entity is spending time in a value-added activity. Initial NVA Cost Initial cost value that will be assigned to the non-value-added cost attribute of the entity. This attribute accrues the costs incurred when an entity is spending time in a non-value-added activity. Initial Waiting Cost Initial cost value that will be assigned to the waiting-cost attribute of the entity. This attribute accrues the costs incurred when an entity is spending time in a wait activity; for example, waiting to be batched or waiting for resource(s) at a Process module. Initial Transfer Cost Initial cost value that will be assigned to the transfer cost attribute of the entity. This attribute accrues the costs incurred when an entity is spending time in a transfer activity. Initial Other Cost Initial cost value that will be assigned to the other cost attribute of the entity. This attribute accrues the costs incurred when an entity is spending time in another activity. Report Statistics Specifies whether or not statistics will be collected automatically and stored in the report database for this entity type.

Definition at line 76 of file Entity.h.

7.42.2 Member Function Documentation

7.42.2.1 _check()

```
bool Entity::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 171 of file Entity.cpp.

7.42.2.2 _loadInstance()

```
bool Entity::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 159 of file Entity.cpp.

7.42.2.3 _saveInstance() `std::map< std::string, std::string > * Entity::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 166 of file Entity.cpp.

7.42.2.4 entityNumber() `Util::identification Entity::entityNumber ( ) const`

Definition at line 154 of file Entity.cpp.

Here is the caller graph for this function:

7.42.2.5 getAttributeValue() [1/4] `double Entity::getAttributeValue ( std::string attributeName )`

Definition at line 88 of file Entity.cpp.

Here is the caller graph for this function:

7.42.2.6 getAttributeValue() [2/4]

```
double Entity::getAttributeValue (
    std::string index,
    std::string attributeName )
```

Definition at line 93 of file Entity.cpp.

Here is the call graph for this function:

7.42.2.7 getAttributeValue() [3/4]

```
double Entity::getAttributeValue (
    std::string index,
    Util::identification attributeID )
```

Definition at line 114 of file Entity.cpp.

Here is the call graph for this function:

7.42.2.8 getAttributeValue() [4/4]

```
double Entity::getAttributeValue (
    Util::identification attributeID )
```

Definition at line 109 of file Entity.cpp.

Here is the call graph for this function:

7.42.2.9 getEntityType() `EntityType * Entity::getEntityType ( ) const`

Definition at line 51 of file Entity.cpp.

Here is the caller graph for this function:

7.42.2.10 getEntityTypeName() `std::string Entity::getEntityTypeName ( ) const`

Definition at line 41 of file Entity.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.42.2.11 setAttributeValue() [1/4]

```
void Entity::setAttributeValue (
    std::string attributeName,
    double value )
```

Definition at line 124 of file Entity.cpp.

Here is the caller graph for this function:

7.42.2.12 setAttributeValue() [2/4]

```
void Entity::setAttributeValue (
    std::string index,
    std::string attributeName,
    double value )
```

Definition at line 128 of file Entity.cpp.

Here is the call graph for this function:

7.42.2.13 setAttributeValue() [3/4]

```
void Entity::setAttributeValue (
    std::string index,
    Util::identification attributeID,
    double value )
```

Definition at line 148 of file Entity.cpp.

Here is the call graph for this function:

7.42.2.14 setAttributeValue() [4/4]

```
void Entity::setAttributeValue (
    Util::identification attributeID,
    double value )
```

Definition at line 143 of file Entity.cpp.

Here is the call graph for this function:

7.42.2.15 setEntityType()

```
void Entity::setEntityType (
    EntityType * entityType )
```

Definition at line 46 of file Entity.cpp.

Here is the caller graph for this function:

7.42.2.16 setEntityTypeName()

```
void Entity::setEntityTypeName (
    std::string entityTypeName ) throw ( )
```

Definition at line 31 of file Entity.cpp.

7.42.2.17 show()

```
std::string Entity::show ( ) [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file Entity.cpp.

Here is the call graph for this function:

7.42.3 Friends And Related Function Documentation

7.42.3.1 Model

```
friend class Model [friend]
```

Definition at line 81 of file Entity.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Entity.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Entity.cpp

7.43 EntityGroup Class Reference

```
#include <EntityGroup.h>
```

Inheritance diagram for EntityGroup:

Collaboration diagram for EntityGroup:

Public Member Functions

- EntityGroup (Model *model, std::string name="")
- virtual ~EntityGroup ()
- virtual std::string show ()
- void insertElement (Entity *element)
- void removeElement (Entity *element)
- unsigned int size ()
- Entity * first ()
- void initBetweenReplications ()

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _createInternalElements ()

Additional Inherited Members

7.43.1 Detailed Description

Definition at line 25 of file EntityGroup.h.

7.43.2 Constructor & Destructor Documentation

7.43.2.1 EntityGroup()

```
EntityGroup::EntityGroup (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file EntityGroup.cpp.

Here is the caller graph for this function:

7.43.2.2 ~EntityGroup()

```
EntityGroup::~EntityGroup ( ) [virtual]
```

Definition at line 21 of file EntityGroup.cpp.

7.43.3 Member Function Documentation

7.43.3.1 _check()

```
bool EntityGroup::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 88 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.2 _createInternalElements() `void EntityGroup::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist before checking such expression

Reimplemented from ModelElement.

Definition at line 97 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.3 _loadInstance() `bool EntityGroup::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 73 of file EntityGroup.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.43.3.4 _saveInstance() `std::map< std::string, std::string > * EntityGroup::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 83 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.5 first() `Entity * EntityGroup::first ( )`

Definition at line 50 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.6 GetPluginInformation() `PluginInformation * EntityGroup::GetPluginInformation ( ) [static]`

Definition at line 58 of file EntityGroup.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.43.3.7 initBetweenReplications() `void EntityGroup::initBetweenReplications ( )`

Definition at line 41 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.8 insertElement() `void EntityGroup::insertElement ( Entity * element )`

Definition at line 30 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.9 LoadInstance()

```
ModelElement * EntityGroup::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 63 of file EntityGroup.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.43.3.10 removeElement()

```
void EntityGroup::removeElement (
    Entity * element )
```

Definition at line 35 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.11 show() `std::string EntityGroup::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 25 of file EntityGroup.cpp.

Here is the call graph for this function:

7.43.3.12 size() `unsigned int EntityGroup::size ( )`

Definition at line 46 of file EntityGroup.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/EntityGroup.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/EntityGroup.cpp

7.44 EntityType Class Reference

```
#include <EntityType.h>
```

Inheritance diagram for EntityType:

Collaboration diagram for EntityType:

Public Member Functions

- EntityType (Model *model, std::string name="")
- virtual ~EntityType ()
- virtual std::string show ()
- void setInitialWaitingCost (double _initialWaitingCost)
- double initialWaitingCost () const
- void setInitialOtherCost (double _initialOtherCost)
- double initialOtherCost () const
- void setInitialNVACost (double _initialNVACost)
- double initialNVACost () const
- void setInitialVACost (double _initialVACost)
- double initialVACost () const
- void setInitialPicture (std::string _initialPicture)
- std::string initialPicture () const
- StatisticsCollector * addGetStatisticsCollector (std::string name)

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _initBetweenReplications ()
- virtual void _createInternalElements ()

Additional Inherited Members

7.44.1 Detailed Description

Definition at line 29 of file EntityType.h.

7.44.2 Constructor & Destructor Documentation

7.44.2.1 EntityType()

```
EntityType::EntityType (
    Model * model,
    std::string name = "" )
```

Definition at line 20 of file EntityType.cpp.

Here is the caller graph for this function:

7.44.2.2 ~EntityType()

```
EntityType::~EntityType ( ) [virtual]
```

Definition at line 33 of file EntityType.cpp.

Here is the call graph for this function:

7.44.3 Member Function Documentation

7.44.3.1 _check() `bool EntityType::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 136 of file EntityType.cpp.

7.44.3.2 _createInternalElements() `void EntityType::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist before checking such expression

Reimplemented from ModelElement.

Definition at line 141 of file EntityType.cpp.

Here is the call graph for this function:

7.44.3.3 _initBetweenReplications() `void EntityType::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 23 of file EntityType.cpp.

Here is the call graph for this function:

7.44.3.4 _loadInstance() `bool EntityType::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 114 of file EntityType.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.44.3.5 _saveInstance() `std::map< std::string, std::string > * EntityType::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 126 of file EntityType.cpp.

Here is the call graph for this function:

7.44.3.6 addGetStatisticsCollector()

```
StatisticsCollector * EntityType::addGetStatisticsCollector
(
            std::string name )
```

Definition at line 85 of file EntityType.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.44.3.7 GetPluginInformation()

```
PluginInformation * EntityType::GetPluginInformation ( ) [static]
```

Definition at line 99 of file EntityType.cpp.

Here is the call graph for this function:

7.44.3.8 initialNVACost() `double EntityType::initialNVACost ( ) const`

Definition at line 65 of file EntityType.cpp.

7.44.3.9 initialOtherCost() `double EntityType::initialOtherCost ( ) const`

Definition at line 57 of file EntityType.cpp.

7.44.3.10 initialPicture() `std::string EntityType::initialPicture ( ) const`

Definition at line 81 of file EntityType.cpp.

7.44.3.11 initialVACost() `double EntityType::initialVACost ( ) const`

Definition at line 73 of file EntityType.cpp.

7.44.3.12 initialWaitingCost() `double EntityType::initialWaitingCost ( ) const`

Definition at line 49 of file EntityType.cpp.

7.44.3.13 LoadInstance()

```
ModelElement * EntityType::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 104 of file EntityType.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.44.3.14 setInitialNVACost()

```
void EntityType::setInitialNVACost (
    double _initialNVACost )
```

Definition at line 61 of file EntityType.cpp.

7.44.3.15 setInitialOtherCost()

```
void EntityType::setInitialOtherCost (
    double _initialOtherCost )
```

Definition at line 53 of file EntityType.cpp.

7.44.3.16 setInitialPicture()

```
void EntityType::setInitialPicture (
    std::string _initialPicture )
```

Definition at line 77 of file EntityType.cpp.

7.44.3.17 setInitialVACost()

```
void EntityType::setInitialVACost (
    double _initialVACost )
```

Definition at line 69 of file EntityType.cpp.

7.44.3.18 setInitialWaitingCost()

```
void EntityType::setInitialWaitingCost (
    double _initialWaitingCost )
```

Definition at line 45 of file EntityType.cpp.

7.44.3.19 show()

```
std::string EntityType::show ( ) [virtual]
```

Reimplemented from ModelElement.

Definition at line 40 of file EntityType.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/EntityType.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/EntityType.cpp

7.45 Event Class Reference

```
#include <Event.h>
```

Collaboration diagram for Event:

Public Member Functions

- Event (double time, Entity *entity, ModelComponent *component, unsigned int componentInputNumber=0)
- Event (double time, Entity *entity, Connection *connection)
- virtual ~Event ()=default
- double getTime () const
- ModelComponent * getComponent () const
- Entity * getEntity () const
- unsigned int getComponentInputNumber () const
- std::string show ()

7.45.1 Detailed Description

An an instantaneaous event, triggered at a certain moment by an entity upon reaching a component. The simulated time advances in discrete points in time and that are the instants that an event is triggered.

Definition at line 28 of file Event.h.

7.45.2 Constructor & Destructor Documentation

7.45.2.1 Event() [1/2]

```
Event::Event (
    double time,
    Entity * entity,
    ModelComponent * component,
    unsigned int componentInputNumber = 0 )
```

Definition at line 18 of file Event.cpp.

7.45.2.2 Event() [2/2]

```
Event::Event (
    double time,
    Entity * entity,
    Connection * connection )
```

Definition at line 25 of file Event.cpp.

7.45.2.3 ~Event()

```
virtual Event::~Event ( ) [virtual], [default]
```

7.45.3 Member Function Documentation

7.45.3.1 getComponent()

```
ModelComponent * Event::getComponent ( ) const
```

Definition at line 50 of file Event.cpp.

7.45.3.2 getComponentInputNumber()

```
unsigned int Event::getComponentInputNumber ( ) const
```

Definition at line 42 of file Event.cpp.

7.45.3.3 getEntity()

```
Entity * Event::getEntity ( ) const
```

Definition at line 54 of file Event.cpp.

Here is the caller graph for this function:

7.45.3.4 getTime() `double Event::getTime ( ) const`

Definition at line 46 of file Event.cpp.

Here is the caller graph for this function:

7.45.3.5 show() `std::string Event::show ( )`

Definition at line 32 of file Event.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Event.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Event.cpp

7.46 Exact Class Reference

```
#include <Exact.h>
```

Collaboration diagram for Exact:

Public Member Functions

- Exact ()
- Exact (int num, int den=1)
- const Exact & operator+ (const Exact &right)
- const Exact & operator+= (const Exact &right)
- const Exact & operator-= (const Exact &right)
- const Exact & operator*= (const Exact &right)
- const Exact & operator/= (const Exact &right)
- const Exact & operator= (const Exact &right)
- const Exact & operator= (const int &right)
- bool operator== (const Exact &right)
- bool operator== (const double &right)
- bool operator< (const Exact &right)
- bool operator< (const double &right)
- bool operator<= (const Exact &right)
- bool operator<= (const double &right)
- bool operator> (const Exact &right)
- bool operator> (const double &right)
- bool operator>= (const Exact &right)
- bool operator>= (const double &right)
- void Simplify ()
- double toFloat ()

7.46.1 Detailed Description

Definition at line 20 of file Exact.h.

7.46.2 Constructor & Destructor Documentation

7.46.2.1 Exact() [1/2] `Exact::Exact ( ) [inline]`

Definition at line 24 of file Exact.h.

7.46.2.2 Exact() [2/2]

```
Exact::Exact (
    int num,
    int den = 1 ) [inline]
```

Definition at line 26 of file Exact.h.

7.46.3 Member Function Documentation

7.46.3.1 operator*=()

```
const Exact& Exact::operator*= (
    const Exact & right ) [inline]
```

Definition at line 44 of file Exact.h.

7.46.3.2 operator+()

```
const Exact& Exact::operator+ (
    const Exact & right ) [inline]
```

Definition at line 30 of file Exact.h.

7.46.3.3 operator+=()

```
const Exact& Exact::operator+= (
    const Exact & right ) [inline]
```

Definition at line 36 of file Exact.h.

7.46.3.4 operator-=()

```
const Exact& Exact::operator-= (
    const Exact & right ) [inline]
```

Definition at line 40 of file Exact.h.

7.46.3.5 operator/=()

```
const Exact& Exact::operator/= (
    const Exact & right ) [inline]
```

Definition at line 48 of file Exact.h.

7.46.3.6 operator<() [1/2]

```
bool Exact::operator< (
    const double & right ) [inline]
```

Definition at line 72 of file Exact.h.

7.46.3.7 operator<() [2/2]

```
bool Exact::operator< (
    const Exact & right ) [inline]
```

Definition at line 68 of file Exact.h.

7.46.3.8 operator<=() [1/2]

```
bool Exact::operator<= (
    const double & right ) [inline]
```

Definition at line 80 of file Exact.h.

7.46.3.9 operator<=() [2/2]

```
bool Exact::operator<= (
    const Exact & right ) [inline]
```

Definition at line 76 of file Exact.h.

7.46.3.10 operator=() [1/2]

```
const Exact& Exact::operator= (
    const Exact & right ) [inline]
```

Definition at line 52 of file Exact.h.

7.46.3.11 operator=() [2/2]

```
const Exact& Exact::operator= (
    const int & right ) [inline]
```

Definition at line 56 of file Exact.h.

7.46.3.12 operator==() [1/2]

```
bool Exact::operator== (
    const double & right ) [inline]
```

Definition at line 64 of file Exact.h.

7.46.3.13 operator==() [2/2]

```
bool Exact::operator== (
    const Exact & right ) [inline]
```

Definition at line 60 of file Exact.h.

7.46.3.14 operator>() [1/2]

```
bool Exact::operator> (
    const double & right ) [inline]
```

Definition at line 88 of file Exact.h.

7.46.3.15 operator>() [2/2]

```
bool Exact::operator> (
    const Exact & right ) [inline]
```

Definition at line 84 of file Exact.h.

7.46.3.16 operator>=() [1/2]

```
bool Exact::operator>= (
    const double & right ) [inline]
```

Definition at line 96 of file Exact.h.

7.46.3.17 operator>=() [2/2]

```
bool Exact::operator>= (
    const Exact & right ) [inline]
```

Definition at line 92 of file Exact.h.

7.46.3.18 Simplify() `void Exact::Simplify ( ) [inline]`

Definition at line 100 of file Exact.h.

7.46.3.19 toFloat() `double Exact::toFloat ( ) [inline]`

Definition at line 102 of file Exact.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/util/Exact.h

7.47 Exit Class Reference

```
#include <Exit.h>
```

Inheritance diagram for Exit:

Collaboration diagram for Exit:

Public Member Functions

- Exit (Model *model, std::string name="")
- virtual ~Exit ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelComponent * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)

Additional Inherited Members

7.47.1 Detailed Description

Exit module DESCRIPTION The Exit module releases the entity's cells on the specified conveyor. If another entity is waiting in queue for the conveyor at the same station when the cells are released, it will then access the conveyor. TYPICAL USES Cases exit a conveyor for packing Bad parts are removed from the conveyor and disposed Passengers remove luggage from the baggage claim conveyor PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Conveyor Name Name of the conveyor on which the entity will exit. If left blank, the previously accessed conveyor is assumed.

of Cells Number of contiguous conveyor cells the entity will relinquish.

Definition at line 37 of file Exit.h.

7.47.2 Constructor & Destructor Documentation

7.47.2.1 Exit()

```
Exit::Exit (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Exit.cpp.

Here is the caller graph for this function:

7.47.2.2 ~Exit()

```
virtual Exit::~Exit ( ) [virtual], [default]
```

7.47.3 Member Function Documentation

7.47.3.1 _check()

```
bool Exit::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Exit.cpp.

7.47.3.2 _execute()

```
void Exit::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Exit.cpp.

Here is the call graph for this function:

7.47.3.3 _initBetweenReplications()

```
void Exit::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 48 of file Exit.cpp.

7.47.3.4 _loadInstance()

```
bool Exit::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 40 of file Exit.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.47.3.5 _saveInstance() `std::map< std::string, std::string > * Exit::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Exit.cpp.

Here is the call graph for this function:

7.47.3.6 GetPluginInformation() `PluginInformation * Exit::GetPluginInformation ( ) [static]`

Definition at line 64 of file Exit.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.47.3.7 LoadInstance()

```
ModelComponent * Exit::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file Exit.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.47.3.8 show()

```
std::string Exit::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 21 of file Exit.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Exit.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Exit.cpp

7.48 ExperimentManager Class Reference

```
#include <ExperimentManager.h>
```

Collaboration diagram for ExperimentManager:

Public Member Functions

- ExperimentManager (Simulator ∗simulator)
- SimulationExperiment ∗ newSimulationExperiment ()
- void insert (SimulationExperiment ∗experiment)
- void remove (SimulationExperiment ∗experiment)
- void setCurrent (SimulationExperiment ∗experiment)
- bool saveSimulationExperiment (std::string filename)
- bool loadSimulationExperiment (std::string filename)
- unsigned int size ()
- SimulationExperiment ∗ front ()
- SimulationExperiment ∗ current ()
- SimulationExperiment ∗ next ()
- List< SimulationExperiment ∗ > ∗ getExperiments () const

7.48.1 Detailed Description

Definition at line 26 of file ExperimentManager.h.

7.48.2 Constructor & Destructor Documentation

7.48.2.1 ExperimentManager() `ExperimentManager::ExperimentManager ( Simulator * simulator )`

Definition at line 17 of file ExperimentManager.cpp.

7.48.3 Member Function Documentation

7.48.3.1 current() `SimulationExperiment * ExperimentManager::current ( )`

Definition at line 78 of file ExperimentManager.cpp.

7.48.3.2 front() `SimulationExperiment * ExperimentManager::front ( )`

Definition at line 82 of file ExperimentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.48.3.3 getExperiments() `List< SimulationExperiment * > * ExperimentManager::getExperiments ( ) const`

Definition at line 22 of file ExperimentManager.cpp.

7.48.3.4 insert()

```
void ExperimentManager::insert (
    SimulationExperiment * experiment )
```

Definition at line 31 of file ExperimentManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.48.3.5 loadSimulationExperiment()

```
bool ExperimentManager::loadSimulationExperiment (
    std::string filename )
```

Definition at line 59 of file ExperimentManager.cpp.

Here is the call graph for this function:

7.48.3.6 newSimulationExperiment()

`SimulationExperiment * ExperimentManager::newSimulation↩ Experiment ( )`

Definition at line 26 of file ExperimentManager.cpp.

7.48.3.7 next()

`SimulationExperiment * ExperimentManager::next ( )`

Definition at line 86 of file ExperimentManager.cpp.

Here is the call graph for this function:

7.48.3.8 remove()

`void ExperimentManager::remove ( SimulationExperiment * experiment )`

Definition at line 37 of file ExperimentManager.cpp.

Here is the call graph for this function:

7.48.3.9 saveSimulationExperiment()

`bool ExperimentManager::saveSimulationExperiment ( std::string filename )`

Definition at line 50 of file ExperimentManager.cpp.

7.48.3.10 setCurrent()

```
void ExperimentManager::setCurrent (
    SimulationExperiment * experiment )
```

Definition at line 74 of file ExperimentManager.cpp.

7.48.3.11 size()

```
unsigned int ExperimentManager::size ( )
```

Definition at line 46 of file ExperimentManager.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ExperimentManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ExperimentManager.cpp

7.49 ExperimentManager_if Class Reference

```
#include <ExperimetManager_if.h>
```

Inheritance diagram for ExperimentManager_if:

Collaboration diagram for ExperimentManager_if:

Public Member Functions

- virtual List< SimulationScenario * > * getScenarios () const =0
- virtual List< SimulationControl * > * getControls () const =0
- virtual List< SimulationResponse * > * getResponses () const =0
- virtual List< SimulationControl * > * extractControlsFromModel (std::string modelFilename) const =0
- virtual List< SimulationResponse * > * extractResponsesFromModel (std::string modelFilename) const =0
- virtual void startSimulationOfScenario (SimulationScenario *scenario)=0
- virtual void startExperiment ()=0
- virtual void stopExperiment ()=0
- virtual void addTraceSimulationHandler (traceSimulationProcessListener traceSimulationProcessListener)=0

7.49.1 Detailed Description

The experiment manager allows to extract controls and responses from a model, include some of then as controls and responses for a set of scenarios to be simulated

Definition at line 26 of file ExperimetManager_if.h.

7.49.2 Member Function Documentation

7.49.2.1 addTraceSimulationHandler()

```
virtual void ExperimentManager_if::addTraceSimulation↩
Handler (
            traceSimulationProcessListener traceSimulationProcessListener ) [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.2 extractControlsFromModel()

```
virtual List<SimulationControl*>* ExperimentManager_if↩
::extractControlsFromModel (
            std::string modelFilename ) const  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.3 extractResponsesFromModel()

```
virtual List<SimulationResponse*>* ExperimentManager_↩
if::extractResponsesFromModel (
            std::string modelFilename ) const  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.4 getControls()

```
virtual List<SimulationControl*>* ExperimentManager_if::getControls ( )
const  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.5 getResponses()

```
virtual List<SimulationResponse*>* ExperimentManager_if::getResponses
( ) const  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.6 getScenarios()

```
virtual List<SimulationScenario*>* ExperimentManager_if::getScenarios
( ) const  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.7 startExperiment()

```
virtual void ExperimentManager_if::startExperiment ( )  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.8 startSimulationOfScenario()

```
virtual void ExperimentManager_if::startSimulationOf↩
Scenario (
            SimulationScenario * scenario )  [pure virtual]
```

Implemented in ExperimentManagerDefaultImpl1.

7.49.2.9 stopExperiment() `virtual void ExperimentManager_if::stopExperiment ( ) [pure virtual]`

Implemented in ExperimentManagerDefaultImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ExperimetManager_if.h

7.50 ExperimentManagerDefaultImpl1 Class Reference

```
#include <ExperimentManagerDefaultImpl1.h>
```

Inheritance diagram for ExperimentManagerDefaultImpl1:

Collaboration diagram for ExperimentManagerDefaultImpl1:

Public Member Functions

- ExperimentManagerDefaultImpl1 ()
- virtual ~ExperimentManagerDefaultImpl1 ()=default
- virtual List< SimulationScenario * > * getScenarios () const
- virtual List< SimulationControl * > * getControls () const
- virtual List< SimulationResponse * > * getResponses () const
- virtual List< SimulationControl * > * extractControlsFromModel (std::string modelFilename) const
- virtual List< SimulationResponse * > * extractResponsesFromModel (std::string modelFilename) const
- virtual void startSimulationOfScenario (SimulationScenario *scenario)
- virtual void startExperiment ()
- virtual void stopExperiment ()
- virtual void addTraceSimulationHandler (traceSimulationProcessListener traceSimulationProcessListener)

7.50.1 Detailed Description

Definition at line 22 of file ExperimentManagerDefaultImpl1.h.

7.50.2 Constructor & Destructor Documentation

7.50.2.1 ExperimentManagerDefaultImpl1() `ExperimentManagerDefaultImpl1::ExperimentManager↩ DefaultImpl1 ( )`

Definition at line 16 of file ExperimentManagerDefaultImpl1.cpp.

7.50.2.2 ~ExperimentManagerDefaultImpl1() `virtual ExperimentManagerDefaultImpl1::~Experiment↩ ManagerDefaultImpl1 ( ) [virtual], [default]`

7.50.3 Member Function Documentation

7.50.3.1 addTraceSimulationHandler() `void ExperimentManagerDefaultImpl1::addTraceSimulation↩ Handler (`

```
            traceSimulationProcessListener traceSimulationProcessListener ) [virtual]
```

Implements ExperimentManager_if.

Definition at line 57 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.2 extractControlsFromModel() `List< SimulationControl * > * ExperimentManagerDefault↩ Impl1::extractControlsFromModel (`

```
            std::string modelFilename ) const [virtual]
```

Implements ExperimentManager_if.

Definition at line 33 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.3 extractResponsesFromModel() `List< SimulationResponse * > * ExperimentManagerDefault↩ Impl1::extractResponsesFromModel (`

```
            std::string modelFilename ) const [virtual]
```

Implements ExperimentManager_if.

Definition at line 39 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.4 getControls() `List< SimulationControl * > * ExperimentManagerDefaultImpl1::getControls ( ) const [virtual]`

Implements ExperimentManager_if.

Definition at line 24 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.5 getResponses() `List< SimulationResponse * > * ExperimentManagerDefaultImpl1::get↩ Responses ( ) const [virtual]`

Implements ExperimentManager_if.

Definition at line 28 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.6 getScenarios() `List< SimulationScenario * > * ExperimentManagerDefaultImpl1::get↩ Scenarios ( ) const [virtual]`

Implements ExperimentManager_if.

Definition at line 19 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.7 startExperiment() `void ExperimentManagerDefaultImpl1::startExperiment ( ) [virtual]`

Implements ExperimentManager_if.

Definition at line 49 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.8 startSimulationOfScenario() `void ExperimentManagerDefaultImpl1::startSimulationOf↩ Scenario ( SimulationScenario * scenario ) [virtual]`

Implements ExperimentManager_if.

Definition at line 45 of file ExperimentManagerDefaultImpl1.cpp.

7.50.3.9 stopExperiment() `void ExperimentManagerDefaultImpl1::stopExperiment ( ) [virtual]`

Implements ExperimentManager_if.

Definition at line 53 of file ExperimentManagerDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.cpp

7.51 Failure Class Reference

```
#include <Failure.h>
```

Inheritance diagram for Failure:

Collaboration diagram for Failure:

Public Member Functions

- Failure (Model ∗model, std::string name="")
- virtual ∼Failure ()=default
- virtual std::string show ()

Static Public Member Functions

- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)
- static PluginInformation ∗ GetPluginInformation ()

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual ParserChangesInformation ∗ _getParserChangesInformation ()

Additional Inherited Members

7.51.1 Detailed Description

Failure module DESCRIPTION The Failure module is designed for use with resources. When a failure occurs, the entire resource (regardless of its capacity) is failed. Failures are designed to be used with single-capacity resources or with multiple-capacity resources whose individual resource units all fail at the same time. TYPICAL USES Breakdown information for a machine Cash register tape refill every "x" customers Random computer shutdowns or restarts PROMPTS Recordset Name of the recordset in the specified file from which to read values. This field is available only if you specify a File Name with a file access type, path, and recordset. Arena uses the Rows and Columns properties to determine the amount of data to read from the recordset. A recordset is required for all file types except .xml. The recordset size must be equal to or greater than the number of rows and columns specified for the expression. Expression Values Lists the value or values of the expression. This property is not available if you specify a File Name from which to read expression values. Expression Value Expression value associated with the expression name. Prompt Description Name The name of the failure associated with one or more resources. Type Determines if the failure is time-based or count-based. Count Defines the number of resource releases for count-based failures. Valid when the Type is Count. Up Time Defines the time between failures for time-based failures. Valid when the Type is Time. Up Time Units Time units for the time between failures (Up Time) for timebased failures. Down Time Defines the duration of the failure. Down Time Units Time units for the duration of the failure (Down Time). Uptime in this State only Defines the state that should be considered for the time between failures (only for time-based failures). If state is not specified, then all states are considered (that is, the time between failures does not depend on the time spent in a specific state, but rather on the total simulation time). For example, you might want to define a failure to be based only on the state Busy, and therefore, the time between downtimes would be based on the amount of time that a resource is busy, not simulated clock time.

Definition at line 67 of file Failure.h.

7.51.2 Constructor & Destructor Documentation

7.51.2.1 Failure()

```
Failure::Failure (
    Model * model,
    std::string name = "" )
```

Definition at line 16 of file Failure.cpp.

Here is the caller graph for this function:

7.51.2.2 ~Failure()

```
virtual Failure::~Failure ( ) [virtual], [default]
```

7.51.3 Member Function Documentation

7.51.3.1 _check()

```
bool Failure::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 58 of file Failure.cpp.

7.51.3.2 _getParserChangesInformation()

```
ParserChangesInformation * Failure::_getParserChanges↩
Information ( ) [protected], [virtual]
```

This method returns all changes in the parser that are needed by plugins of this ModelElements. When connecting a new plugin, ParserChangesInformation are used to change parser source code, whch is after compiled and dinamically linked to to simulator kernel to reflect the changes

Reimplemented from ModelElement.

Definition at line 65 of file Failure.cpp.

7.51.3.3 _loadInstance()

```
bool Failure::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 39 of file Failure.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.51.3.4 _saveInstance() `std::map< std::string, std::string > * Failure::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 51 of file Failure.cpp.

Here is the call graph for this function:

7.51.3.5 GetPluginInformation() `PluginInformation * Failure::GetPluginInformation ( ) [static]`

Definition at line 24 of file Failure.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.51.3.6 LoadInstance()

```
ModelElement * Failure::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 29 of file Failure.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.51.3.7 show()

```
std::string Failure::show ( ) [virtual]
```

Reimplemented from ModelElement.

Definition at line 19 of file Failure.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Failure.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Failure.cpp

7.52 File Class Reference

```
#include <File.h>
```

Inheritance diagram for File:

Collaboration diagram for File:

Public Member Functions

- File (Model *model, std::string name="")
- virtual ~File ()=default
- virtual std::string show ()

Static Public Member Functions

- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields)
- static PluginInformation * GetPluginInformation ()

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual ParserChangesInformation * _getParserChangesInformation ()

Additional Inherited Members

7.52.1 Detailed Description

File module DESCRIPTION Use the File module to access external files for the ReadWrite module, Variable module, and Expression module. The File module identifies the system file name and defines the access type and operational characteristics of the file. TYPICAL USES File containing predefined airline flight data File specifying customer order times and relevant information File to write user model configuration data from menu input PROMPTS Prompt Description Name The name of the file whose characteristics are being defined. This name must be unique. Access Type The file type. Operating System File Name Name of the actual file that is being read from or to which it is being written. Connecting String Connection string used to open ADO connection to the data source. Structure File structure, which can be unformatted, free format, or a specific C or FORTRAN format. End of File Action Type of action to occur if an end of file condition is reached. Initialize Option Action to be taken on file at beginning of each simulation replication. Comment Character Character indicating comment record. Recordset Name Name used to identify the recordset in the Expression, ReadWrite, and Variable modules. This name must be unique within the file. This field is available for Microsoft Excel, Microsoft Excel 2007, Microsoft Access, Microsoft Access 2007, and ActiveX Data Objects files. CommandText Text of the command that will be used to open the recordset (for example, SQL statement, procedure name, table name.) This field is available for ActiveX Data Object files only. CommandType Type of command entered in the CommandText. Named Range The named range in the Excel workbook to which the recordset refers. Table Name The name of the table in the Access database to which the recordset refers.

Definition at line 64 of file File.h.

7.52.2 Constructor & Destructor Documentation

7.52.2.1 File()

```
File::File (
    Model * model,
    std::string name = "" )
```

Definition at line 16 of file File.cpp.

Here is the caller graph for this function:

7.52.2.2 ~File()

```
virtual File::~File ( ) [virtual], [default]
```

7.52.3 Member Function Documentation

7.52.3.1 _check()

```
bool File::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 59 of file File.cpp.

7.52.3.2 _getParserChangesInformation()

```
ParserChangesInformation * File::_getParserChanges↩
Information ( ) [protected], [virtual]
```

This method returns all changes in the parser that are needed by plugins of this ModelElements. When connecting a new plugin, ParserChangesInformation are used to change parser source code, whch is after compiled and dinamically linked to to simulator kernel to reflect the changes

Reimplemented from ModelElement.

Definition at line 66 of file File.cpp.

7.52.3.3 _loadInstance()

```
bool File::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 40 of file File.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.52.3.4 _saveInstance() `std::map< std::string, std::string > * File::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 52 of file File.cpp.

Here is the call graph for this function:

7.52.3.5 GetPluginInformation() `PluginInformation * File::GetPluginInformation ( ) [static]`

Definition at line 25 of file File.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.52.3.6 LoadInstance()

```
ModelElement * File::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 30 of file File.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.52.3.7 show()

```
std::string File::show ( ) [virtual]
```

Reimplemented from ModelElement.

Definition at line 20 of file File.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/File.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/File.cpp

7.53 Ui::formPlugins Class Reference

```
#include <ui_formPlugins.h>
```

Inheritance diagram for Ui::formPlugins:

Collaboration diagram for Ui::formPlugins:

Additional Inherited Members

7.53.1 Detailed Description

Definition at line 50 of file ui_formPlugins.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/ui_formPlugins.h

7.54 FullSimulationOfComplexModel Class Reference

```
#include <FullSimulationOfComplexModel.h>
```

Inheritance diagram for FullSimulationOfComplexModel:

Collaboration diagram for FullSimulationOfComplexModel:

Public Member Functions

- FullSimulationOfComplexModel ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.54.1 Detailed Description

Definition at line 19 of file FullSimulationOfComplexModel.h.

7.54.2 Constructor & Destructor Documentation

7.54.2.1 FullSimulationOfComplexModel() `FullSimulationOfComplexModel::FullSimulationOfComplex↩ Model ( )`

Definition at line 28 of file FullSimulationOfComplexModel.cpp.

7.54.3 Member Function Documentation

7.54.3.1 main()

```
int FullSimulationOfComplexModel::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 36 of file FullSimulationOfComplexModel.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/FullSimulationOfComplexModel.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/FullSimulationOfComplexModel.cpp

7.55 ParserManager::GenerateNewParserResult Struct Reference

```
#include <ParserManager.h>
```

Collaboration diagram for ParserManager::GenerateNewParserResult:

Public Attributes

- bool result
- std::string bisonMessages
- std::string lexMessages
- std::string compilationMessages
- NewParser newParser

7.55.1 Detailed Description

Definition at line 30 of file ParserManager.h.

7.55.2 Member Data Documentation

7.55.2.1 bisonMessages `std::string ParserManager::GenerateNewParserResult::bisonMessages`

Definition at line 32 of file ParserManager.h.

7.55.2.2 compilationMessages `std::string ParserManager::GenerateNewParserResult::compilationMessages`

Definition at line 34 of file ParserManager.h.

7.55.2.3 lexMessages `std::string ParserManager::GenerateNewParserResult::lexMessages`

Definition at line 33 of file ParserManager.h.

7.55.2.4 newParser `NewParser ParserManager::GenerateNewParserResult::newParser`

Definition at line 35 of file ParserManager.h.

7.55.2.5 result `bool ParserManager::GenerateNewParserResult::result`

Definition at line 31 of file ParserManager.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.h

7.56 GenesysApplication_if Class Reference

`#include <GenesysApplication_if.h>`

Inheritance diagram for GenesysApplication_if:

Collaboration diagram for GenesysApplication_if:

Public Member Functions

- virtual int main (int argc, char ∗∗argv)=0

7.56.1 Detailed Description

Definition at line 17 of file GenesysApplication_if.h.

7.56.2 Member Function Documentation

7.56.2.1 main()

```
virtual int GenesysApplication_if::main (
    int argc,
    char ** argv ) [pure virtual]
```

Implemented in BaseConsoleGenesysApplication, OperatingSystem03, OperatingSystem02, FullSimulationOfComplexModel, AnElectronicAssemblyAndTestSystem, Smart_Sequence, Smart_SeizeDelayReleaseMany, Smart_SeizeDelayRelease, Smart_RouteStation, Smart_ProcessSet, Smart_Process, Smart_ParserModelFunctions, Smart_Parser, Smart_OnEvent, Smart_ModelInfoModelSimulation, Smart_HoldSignal, Smart_Dummy, Smart_Delay, Smart_AssignWriteSeizes, Book_Cap02_Example01, GenesysTerminalApp, and GenesysQtGUI.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/GenesysApplication_if.h

7.57 genesyspp_driver Class Reference

```
#include <Genesys++-driver.h>
```

Collaboration diagram for genesyspp_driver:

Public Member Functions

- genesyspp_driver ()
- genesyspp_driver (Model ∗model, Sampler_if ∗sampler, bool throws=false)
- virtual ∼genesyspp_driver ()=default
- void scan_begin_file ()
- void scan_end_file ()
- void scan_begin_str ()
- void scan_end_str ()
- int parse_file (const std::string &f)
- int parse_str (const std::string &str)
- void error (const yy::location &l, const std::string &m)
- void error (const std::string &m)
- double getResult ()
- void setResult (double value)
- bool getThrowsException ()
- void setThrowsException (bool throws)
- void setErrorMessage (std::string message)
- std::string getErrorMessage ()
- Model ∗ getModel ()
- std::string getFile ()
- void setFile (std::string f)
- std::string getStrToParse ()
- void setStrToParse (std::string str)
- void setSampler (Sampler_if ∗_sampler)
- Sampler_if ∗ sampler () const

7.57.1 Detailed Description

Definition at line 25 of file Genesys++-driver.h.

7.57.2 Constructor & Destructor Documentation

7.57.2.1 genesyspp_driver() [1/2] genesyspp_driver::genesyspp_driver ( )

Definition at line 7 of file Genesys++-driver.cpp.

7.57.2.2 genesyspp_driver() [2/2]

```
genesyspp_driver::genesyspp_driver (
    Model * model,
    Sampler_if * sampler,
    bool throws = false )
```

Definition at line 10 of file Genesys++-driver.cpp.

Here is the call graph for this function:

7.57.2.3 ~genesyspp_driver() virtual genesyspp_driver::~genesyspp_driver ( ) [virtual], [default]

7.57.3 Member Function Documentation

7.57.3.1 error() [1/2]

```
void genesyspp_driver::error (
    const std::string & m )
```

Definition at line 118 of file Genesys++-driver.cpp.

Here is the call graph for this function:

7.57.3.2 error() [2/2]

```
void genesyspp_driver::error (
    const yy::location & l,
    const std::string & m )
```

Definition at line 109 of file Genesys++-driver.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.57.3.3 getErrorMessage() `std::string genesyspp_driver::getErrorMessage ( )`

Definition at line 76 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.4 getFile() `std::string genesyspp_driver::getFile ( )`

Definition at line 84 of file Genesys++-driver.cpp.

7.57.3.5 getModel() `Model * genesyspp_driver::getModel ( )`

Definition at line 80 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.6 getResult() `double genesyspp_driver::getResult ( )`

Definition at line 60 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.7 getStrToParse() `std::string genesyspp_driver::getStrToParse ( )`

Definition at line 92 of file Genesys++-driver.cpp.

7.57.3.8 getThrowsException() `bool genesyspp_driver::getThrowsException ( )`

Definition at line 64 of file Genesys++-driver.cpp.

7.57.3.9 parse_file() `int genesyspp_driver::parse_file ( const std::string & f )`

Definition at line 34 of file Genesys++-driver.cpp.

Here is the call graph for this function:

7.57.3.10 parse_str()

```
int genesyspp_driver::parse_str (
    const std::string & str )
```

Definition at line 45 of file Genesys++-driver.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.57.3.11 sampler()

```
Sampler_if * genesyspp_driver::sampler ( ) const
```

Definition at line 104 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.12 scan_begin_file() `void genesyspp_driver::scan_begin_file ( )`

Definition at line 6803 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.57.3.13 scan_begin_str() `void genesyspp_driver::scan_begin_str ( )`

Definition at line 6815 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.57.3.14 scan_end_file() `void genesyspp_driver::scan_end_file ( )`

Definition at line 6829 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

7.57.3.15 scan_end_str() `void genesyspp_driver::scan_end_str ( )`

Definition at line 6835 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.57.3.16 setErrorMessage() `void genesyspp_driver::setErrorMessage ( std::string message )`

Definition at line 72 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.17 setFile() `void genesyspp_driver::setFile ( std::string f )`

Definition at line 88 of file Genesys++-driver.cpp.

7.57.3.18 setResult() `void genesyspp_driver::setResult ( double value )`

Definition at line 56 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.19 setSampler() `void genesyspp_driver::setSampler ( Sampler_if * _sampler )`

Definition at line 100 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

7.57.3.20 setStrToParse()

```
void genesyspp_driver::setStrToParse (
    std::string str )
```

Definition at line 96 of file Genesys++-driver.cpp.

7.57.3.21 setThrowsException()

```
void genesyspp_driver::setThrowsException (
    bool throws )
```

Definition at line 68 of file Genesys++-driver.cpp.

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-driver.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-driver.cpp
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/Genesys++-scanner.cpp

7.58 yy::genesyspp_parser Class Reference

A Bison parser.

```
#include <GenesysParser.h>
```

Collaboration diagram for yy::genesyspp_parser:

Classes

- struct basic_symbol
- struct by_kind

  Type access provider for token (enum) based symbols.
- class context
- class semantic_type
- struct symbol_kind

  Symbol kinds.
- struct symbol_type

  "External" symbols: returned by the scanner.

- struct syntax_error

  Syntax errors thrown from user actions.
- struct token

  Token kinds.

Public Types

- typedef location location_type

  Symbol locations.
- typedef token::yytokentype token_kind_type

  Token kind, as returned by yylex.
- typedef token_kind_type token_type

  Backward compatibility alias (Bison 3.6).
- typedef symbol_kind::symbol_kind_type symbol_kind_type

  (Internal) symbol kind.
- typedef by_kind by_type

  Backward compatibility for a private implementation detail (Bison 3.6).
- typedef int debug_level_type

  Type for debugging levels.

Public Member Functions

- genesyspp_parser (genesyspp_driver &driver_yyarg)

  Build a parser object.
- virtual ~genesyspp_parser ()
- int operator() ()
- virtual int parse ()
- std::ostream & debug_stream () const YY_ATTRIBUTE_PURE

  The current debugging stream.
- void set_debug_stream (std::ostream &)

  Set the current debugging stream.
- debug_level_type debug_level () const YY_ATTRIBUTE_PURE

  The current debugging level.
- void set_debug_level (debug_level_type l)

  Set the current debugging level.
- virtual void error (const location_type &loc, const std::string &msg)
- void error (const syntax_error &err)

  Report a syntax error.

Static Public Member Functions

- static std::string symbol_name (symbol_kind_type yysymbol)
- static symbol_type make_END (const location_type &l)
- static symbol_type make_YYerror (const location_type &l)
- static symbol_type make_YYUNDEF (const location_type &l)
- static symbol_type make_NUMD (const obj_t &v, const location_type &l)
- static symbol_type make_NUMH (const obj_t &v, const location_type &l)
- static symbol_type make_CTEZERO (const obj_t &v, const location_type &l)
- static symbol_type make_oLE (const obj_t &v, const location_type &l)
- static symbol_type make_oGE (const obj_t &v, const location_type &l)

- static symbol_type make_oEQ (const obj_t &v, const location_type &l)
- static symbol_type make_oNE (const obj_t &v, const location_type &l)
- static symbol_type make_oAND (const obj_t &v, const location_type &l)
- static symbol_type make_oOR (const obj_t &v, const location_type &l)
- static symbol_type make_oNAND (const obj_t &v, const location_type &l)
- static symbol_type make_oXOR (const obj_t &v, const location_type &l)
- static symbol_type make_oNOT (const obj_t &v, const location_type &l)
- static symbol_type make_fSIN (const obj_t &v, const location_type &l)
- static symbol_type make_fCOS (const obj_t &v, const location_type &l)
- static symbol_type make_fROUND (const obj_t &v, const location_type &l)
- static symbol_type make_fMOD (const obj_t &v, const location_type &l)
- static symbol_type make_fTRUNC (const obj_t &v, const location_type &l)
- static symbol_type make_fFRAC (const obj_t &v, const location_type &l)
- static symbol_type make_fEXP (const obj_t &v, const location_type &l)
- static symbol_type make_fSQRT (const obj_t &v, const location_type &l)
- static symbol_type make_fLOG (const obj_t &v, const location_type &l)
- static symbol_type make_fLN (const obj_t &v, const location_type &l)
- static symbol_type make_fVAL (const obj_t &v, const location_type &l)
- static symbol_type make_fEVAL (const obj_t &v, const location_type &l)
- static symbol_type make_fLENG (const obj_t &v, const location_type &l)
- static symbol_type make_fRND1 (const obj_t &v, const location_type &l)
- static symbol_type make_fEXPO (const obj_t &v, const location_type &l)
- static symbol_type make_fNORM (const obj_t &v, const location_type &l)
- static symbol_type make_fUNIF (const obj_t &v, const location_type &l)
- static symbol_type make_fWEIB (const obj_t &v, const location_type &l)
- static symbol_type make_fLOGN (const obj_t &v, const location_type &l)
- static symbol_type make_fGAMM (const obj_t &v, const location_type &l)
- static symbol_type make_fERLA (const obj_t &v, const location_type &l)
- static symbol_type make_fTRIA (const obj_t &v, const location_type &l)
- static symbol_type make_fBETA (const obj_t &v, const location_type &l)
- static symbol_type make_fDISC (const obj_t &v, const location_type &l)
- static symbol_type make_fTNOW (const obj_t &v, const location_type &l)
- static symbol_type make_fTFIN (const obj_t &v, const location_type &l)
- static symbol_type make_fMAXREP (const obj_t &v, const location_type &l)
- static symbol_type make_fNUMREP (const obj_t &v, const location_type &l)
- static symbol_type make_fIDENT (const obj_t &v, const location_type &l)
- static symbol_type make_cIF (const obj_t &v, const location_type &l)
- static symbol_type make_cELSE (const obj_t &v, const location_type &l)
- static symbol_type make_cFOR (const obj_t &v, const location_type &l)
- static symbol_type make_cTO (const obj_t &v, const location_type &l)
- static symbol_type make_cDO (const obj_t &v, const location_type &l)
- static symbol_type make_ATRIB (const obj_t &v, const location_type &l)
- static symbol_type make_CSTAT (const obj_t &v, const location_type &l)
- static symbol_type make_fTAVG (const obj_t &v, const location_type &l)
- static symbol_type make_ILLEGAL (const obj_t &v, const location_type &l)
- static symbol_type make_RESOURCE (const obj_t &v, const location_type &l)
- static symbol_type make_fNR (const obj_t &v, const location_type &l)
- static symbol_type make_fMR (const obj_t &v, const location_type &l)
- static symbol_type make_fIRF (const obj_t &v, const location_type &l)
- static symbol_type make_fRESSEIZES (const obj_t &v, const location_type &l)
- static symbol_type make_fSTATE (const obj_t &v, const location_type &l)
- static symbol_type make_fSETSUM (const obj_t &v, const location_type &l)
- static symbol_type make_fRESUTIL (const obj_t &v, const location_type &l)
- static symbol_type make_QUEUE (const obj_t &v, const location_type &l)
- static symbol_type make_fNQ (const obj_t &v, const location_type &l)

- static symbol_type make_fFIRSTINQ (const obj_t &v, const location_type &l)
- static symbol_type make_fLASTINQ (const obj_t &v, const location_type &l)
- static symbol_type make_fSAQUE (const obj_t &v, const location_type &l)
- static symbol_type make_fAQUE (const obj_t &v, const location_type &l)
- static symbol_type make_fENTATRANK (const obj_t &v, const location_type &l)
- static symbol_type make_SET (const obj_t &v, const location_type &l)
- static symbol_type make_fNUMSET (const obj_t &v, const location_type &l)
- static symbol_type make_VARI (const obj_t &v, const location_type &l)
- static symbol_type make_FORM (const obj_t &v, const location_type &l)
- static symbol_type make_fNUMGR (const obj_t &v, const location_type &l)
- static symbol_type make_fATRGR (const obj_t &v, const location_type &l)
- static symbol_type make_LPAREN (const location_type &l)
- static symbol_type make_RPAREN (const location_type &l)
- static symbol_type make_LBRACKET (const location_type &l)
- static symbol_type make_RBRACKET (const location_type &l)
- static symbol_type make_PLUS (const location_type &l)
- static symbol_type make_MINUS (const location_type &l)
- static symbol_type make_STAR (const location_type &l)
- static symbol_type make_POWER (const location_type &l)
- static symbol_type make_SLASH (const location_type &l)
- static symbol_type make_LESS (const location_type &l)
- static symbol_type make_GREATER (const location_type &l)
- static symbol_type make_ASSIGN (const location_type &l)
- static symbol_type make_COMMA (const location_type &l)
- static symbol_type make_NEG (const location_type &l)

Static Public Attributes

- static const symbol_kind_type YYNTOKENS = symbol_kind::YYNTOKENS

  The number of tokens.

7.58.1 Detailed Description

A Bison parser.

Definition at line 222 of file GenesysParser.h.

7.58.2 Member Typedef Documentation

7.58.2.1 by_type `typedef by_kind yy::genesyspp_parser::by_type`

Backward compatibility for a private implementation detail (Bison 3.6).

Definition at line 1148 of file GenesysParser.h.

7.58.2.2 debug_level_type `typedef int yy::genesyspp_parser::debug_level_type`

Type for debugging levels.

Definition at line 1212 of file GenesysParser.h.

7.58.2.3 location_type `typedef location yy::genesyspp_parser::location_type`

Symbol locations.

Definition at line 545 of file GenesysParser.h.

7.58.2.4 symbol_kind_type `typedef symbol_kind::symbol_kind_type yy::genesyspp_parser::symbol_kind_type`

(Internal) symbol kind.

Definition at line 793 of file GenesysParser.h.

7.58.2.5 token_kind_type `typedef token::yytokentype yy::genesyspp_parser::token_kind_type`

Token kind, as returned by yylex.

Definition at line 665 of file GenesysParser.h.

7.58.2.6 token_type `typedef token_kind_type yy::genesyspp_parser::token_type`

Backward compatibility alias (Bison 3.6).

Definition at line 668 of file GenesysParser.h.

7.58.3 Constructor & Destructor Documentation

7.58.3.1 genesyspp_parser() `yy::genesyspp_parser::genesyspp_parser ( genesyspp_driver & driver_yyarg )`

Build a parser object.

Definition at line 145 of file GenesysParser.cpp.

7.58.3.2 ~genesyspp_parser() `yy::genesyspp_parser::~genesyspp_parser ( ) [virtual]`

Definition at line 155 of file GenesysParser.cpp.

7.58.4 Member Function Documentation

7.58.4.1 debug_level() `genesyspp_parser::debug_level_type yy::genesyspp_parser::debug_level ( ) const`

The current debugging level.

Definition at line 714 of file GenesysParser.cpp.

7.58.4.2 debug_stream() `std::ostream & yy::genesyspp_parser::debug_stream ( ) const`

The current debugging stream.

Definition at line 701 of file GenesysParser.cpp.

7.58.4.3 error() [1/2]

```
void yy::genesyspp_parser::error (
    const location_type & loc,
    const std::string & msg ) [virtual]
```

Report a syntax error.

Parameters

loc	where the syntax error is found.
msg	a description of the syntax error.

Definition at line 2712 of file GenesysParser.cpp.

Here is the caller graph for this function:

7.58.4.4 error() [2/2]

```
void yy::genesyspp_parser::error (
    const syntax_error & err )
```

Report a syntax error.

Definition at line 1919 of file GenesysParser.cpp.

Here is the call graph for this function:

7.58.4.5 make_ASSIGN()

```
static symbol_type yy::genesyspp_parser::make_ASSIGN (
    const location_type & l )  [inline], [static]
```

Definition at line 2517 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.6 make_ATRIB()

```
static symbol_type yy::genesyspp_parser::make_ATRIB (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 1977 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.7 make_cDO()

```
static symbol_type yy::genesyspp_parser::make_cDO (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1962 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.8 make_cELSE()

```
static symbol_type yy::genesyspp_parser::make_cELSE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1917 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.9 make_cFOR()

```
static symbol_type yy::genesyspp_parser::make_cFOR (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1932 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.10 make_cIF()

```
static symbol_type yy::genesyspp_parser::make_cIF (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1902 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.11 make_COMMA()

```
static symbol_type yy::genesyspp_parser::make_COMMA (
    const location_type & l ) [inline], [static]
```

Definition at line 2532 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.12 make_CSTAT()

```
static symbol_type yy::genesyspp_parser::make_CSTAT (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1992 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.13 make_CTEZERO()

```
static symbol_type yy::genesyspp_parser::make_CTEZERO (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1317 of file GenesysParser.h.

7.58.4.14 make_cTO()

```
static symbol_type yy::genesyspp_parser::make_cTO (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1947 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.15 make_END()

```
static symbol_type yy::genesyspp_parser::make_END (
    const location_type & l ) [inline], [static]
```

Definition at line 1242 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.16 make_fAQUE()

```
static symbol_type yy::genesyspp_parser::make_fAQUE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2232 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.17 make_fATRGR()

```
static symbol_type yy::genesyspp_parser::make_fATRGR (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2337 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.18 make_fBETA()

```
static symbol_type yy::genesyspp_parser::make_fBETA (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1797 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.19 make_fCOS()

```
static symbol_type yy::genesyspp_parser::make_fCOS (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1482 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.20 make_fDISC()

```
static symbol_type yy::genesyspp_parser::make_fDISC (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1812 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.21 make_fENTATRANK()

```
static symbol_type yy::genesyspp_parser::make_fENTATRANK (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2247 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.22 make_fERLA()

```
static symbol_type yy::genesyspp_parser::make_fERLA (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1767 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.23 make_fEVAL()

```
static symbol_type yy::genesyspp_parser::make_fEVAL (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1632 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.24 make_fEXP()

```
static symbol_type yy::genesyspp_parser::make_fEXP (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1557 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.25 make_fEXPO()

```
static symbol_type yy::genesyspp_parser::make_fEXPO (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1677 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.26 make_fFIRSTINQ()

```
static symbol_type yy::genesyspp_parser::make_fFIRSTINQ (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2187 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.27 make_fFRAC()

```
static symbol_type yy::genesyspp_parser::make_fFRAC (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1542 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.28 make_fGAMM()

```
static symbol_type yy::genesyspp_parser::make_fGAMM (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1752 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.29 make_fIDENT()

```
static symbol_type yy::genesyspp_parser::make_fIDENT (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1887 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.30 make_fIRF()

```
static symbol_type yy::genesyspp_parser::make_fIRF (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2082 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.31 make_fLASTINQ()

```
static symbol_type yy::genesyspp_parser::make_fLASTINQ (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2202 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.32 make_fLENG()

```
static symbol_type yy::genesyspp_parser::make_fLENG (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1647 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.33 make_fLN()

```
static symbol_type yy::genesyspp_parser::make_fLN (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1602 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.34 make_fLOG()

```
static symbol_type yy::genesyspp_parser::make_fLOG (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1587 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.35 make_fLOGN()

```
static symbol_type yy::genesyspp_parser::make_fLOGN (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1737 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.36 make_fMAXREP()

```
static symbol_type yy::genesyspp_parser::make_fMAXREP (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1857 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.37 make_fMOD()

```
static symbol_type yy::genesyspp_parser::make_fMOD (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1512 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.38 make_fMR()

```
static symbol_type yy::genesyspp_parser::make_fMR (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2067 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.39 make_fNORM()

```
static symbol_type yy::genesyspp_parser::make_fNORM (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1692 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.40 make_fNQ()

```
static symbol_type yy::genesyspp_parser::make_fNQ (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 2172 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.41 make_fNR()

```
static symbol_type yy::genesyspp_parser::make_fNR (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 2052 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.42 make_fNUMGR()

```
static symbol_type yy::genesyspp_parser::make_fNUMGR (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 2322 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.43 make_fNUMREP()

```
static symbol_type yy::genesyspp_parser::make_fNUMREP (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1872 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.44 make_fNUMSET()

```
static symbol_type yy::genesyspp_parser::make_fNUMSET (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2277 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.45 make_FORM()

```
static symbol_type yy::genesyspp_parser::make_FORM (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2307 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.46 make_fRESSEIZES()

```
static symbol_type yy::genesyspp_parser::make_fRESSEIZES (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2097 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.47 make_fRESUTIL()

```
static symbol_type yy::genesyspp_parser::make_fRESUTIL (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2142 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.48 make_fRND1()

```
static symbol_type yy::genesyspp_parser::make_fRND1 (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1662 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.49 make_fROUND()

```
static symbol_type yy::genesyspp_parser::make_fROUND (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1497 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.50 make_fSAQUE()

```
static symbol_type yy::genesyspp_parser::make_fSAQUE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2217 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.51 make_fSETSUM()

```
static symbol_type yy::genesyspp_parser::make_fSETSUM (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2127 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.52 make_fSIN()

```
static symbol_type yy::genesyspp_parser::make_fSIN (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1467 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.53 make_fSQRT()

```
static symbol_type yy::genesyspp_parser::make_fSQRT (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1572 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.54 make_fSTATE()

```
static symbol_type yy::genesyspp_parser::make_fSTATE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2112 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.55 make_fTAVG()

```
static symbol_type yy::genesyspp_parser::make_fTAVG (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2007 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.56 make_fTFIN()

```
static symbol_type yy::genesyspp_parser::make_fTFIN (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1842 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.57 make_fTNOW()

```
static symbol_type yy::genesyspp_parser::make_fTNOW (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1827 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.58 make_fTRIA()

```
static symbol_type yy::genesyspp_parser::make_fTRIA (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1782 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.59 make_fTRUNC()

```
static symbol_type yy::genesyspp_parser::make_fTRUNC (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1527 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.60 make_fUNIF()

```
static symbol_type yy::genesyspp_parser::make_fUNIF (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1707 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.61 make_fVAL()

```
static symbol_type yy::genesyspp_parser::make_fVAL (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1617 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.62 make_fWEIB()

```
static symbol_type yy::genesyspp_parser::make_fWEIB (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1722 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.63 make_GREATER()

```
static symbol_type yy::genesyspp_parser::make_GREATER (
    const location_type & l ) [inline], [static]
```

Definition at line 2502 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.64 make_ILLEGAL()

```
static symbol_type yy::genesyspp_parser::make_ILLEGAL (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2022 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.65 make_LBRACKET()

```
static symbol_type yy::genesyspp_parser::make_LBRACKET (
    const location_type & l ) [inline], [static]
```

Definition at line 2382 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.66 make_LESS()

```
static symbol_type yy::genesyspp_parser::make_LESS (
    const location_type & l ) [inline], [static]
```

Definition at line 2487 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.67 make_LPAREN()

```
static symbol_type yy::genesyspp_parser::make_LPAREN (
    const location_type & l ) [inline], [static]
```

Definition at line 2352 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.68 make_MINUS()

```
static symbol_type yy::genesyspp_parser::make_MINUS (
    const location_type & l ) [inline], [static]
```

Definition at line 2427 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.69 make_NEG()

```
static symbol_type yy::genesyspp_parser::make_NEG (
    const location_type & l )  [inline], [static]
```

Definition at line 2547 of file GenesysParser.h.

7.58.4.70 make_NUMD()

```
static symbol_type yy::genesyspp_parser::make_NUMD (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 1287 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.71 make_NUMH()

```
static symbol_type yy::genesyspp_parser::make_NUMH (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 1302 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.72 make_oAND()

```
static symbol_type yy::genesyspp_parser::make_oAND (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1392 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.73 make_oEQ()

```
static symbol_type yy::genesyspp_parser::make_oEQ (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1362 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.74 make_oGE()

```
static symbol_type yy::genesyspp_parser::make_oGE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1347 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.75 make_oLE()

```
static symbol_type yy::genesyspp_parser::make_oLE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1332 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.76 make_oNAND()

```
static symbol_type yy::genesyspp_parser::make_oNAND (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1422 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.77 make_oNE()

```
static symbol_type yy::genesyspp_parser::make_oNE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1377 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.78 make_oNOT()

```
static symbol_type yy::genesyspp_parser::make_oNOT (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1452 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.79 make_oOR()

```
static symbol_type yy::genesyspp_parser::make_oOR (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1407 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.80 make_oXOR()

```
static symbol_type yy::genesyspp_parser::make_oXOR (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 1437 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.81 make_PLUS()

```
static symbol_type yy::genesyspp_parser::make_PLUS (
    const location_type & l )  [inline], [static]
```

Definition at line 2412 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.82 make_POWER()

```
static symbol_type yy::genesyspp_parser::make_POWER (
    const location_type & l )  [inline], [static]
```

Definition at line 2457 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.83 make_QUEUE()

```
static symbol_type yy::genesyspp_parser::make_QUEUE (
    const obj_t & v,
    const location_type & l )  [inline], [static]
```

Definition at line 2157 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.84 make_RBRACKET()

```
static symbol_type yy::genesyspp_parser::make_RBRACKET (
    const location_type & l ) [inline], [static]
```

Definition at line 2397 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.85 make_RESOURCE()

```
static symbol_type yy::genesyspp_parser::make_RESOURCE (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2037 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.86 make_RPAREN()

```
static symbol_type yy::genesyspp_parser::make_RPAREN (
    const location_type & l ) [inline], [static]
```

Definition at line 2367 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.87 make_SET()

```
static symbol_type yy::genesyspp_parser::make_SET (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2262 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.88 make_SLASH()

```
static symbol_type yy::genesyspp_parser::make_SLASH (
    const location_type & l ) [inline], [static]
```

Definition at line 2472 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.89 make_STAR()

```
static symbol_type yy::genesyspp_parser::make_STAR (
    const location_type & l ) [inline], [static]
```

Definition at line 2442 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.90 make_VARI()

```
static symbol_type yy::genesyspp_parser::make_VARI (
    const obj_t & v,
    const location_type & l ) [inline], [static]
```

Definition at line 2292 of file GenesysParser.h.

Here is the caller graph for this function:

7.58.4.91 make_YYerror()

```
static symbol_type yy::genesyspp_parser::make_YYerror (
    const location_type & l ) [inline], [static]
```

Definition at line 1257 of file GenesysParser.h.

7.58.4.92 make_YYUNDEF()

```
static symbol_type yy::genesyspp_parser::make_YYUNDEF (
    const location_type & l ) [inline], [static]
```

Definition at line 1272 of file GenesysParser.h.

7.58.4.93 operator()()

```
int yy::genesyspp_parser::operator() ( )
```

Parse. An alias for parse ().

Returns

0 iff parsing succeeded.

Definition at line 749 of file GenesysParser.cpp.

Here is the call graph for this function:

7.58.4.94 parse() `int yy::genesyspp_parser::parse ( ) [virtual]`

Parse.

Returns

0 iff parsing succeeded.

Length of the RHS of the rule being reduced.

The lookahead symbol.

The locations where the error started and ended.

The return value of parse ().

Definition at line 755 of file GenesysParser.cpp.

Here is the caller graph for this function:

7.58.4.95 set_debug_level()

```
void yy::genesyspp_parser::set_debug_level (
    debug_level_type l )
```

Set the current debugging level.

Definition at line 720 of file GenesysParser.cpp.

7.58.4.96 set_debug_stream()

```
void yy::genesyspp_parser::set_debug_stream (
    std::ostream & o )
```

Set the current debugging stream.

Definition at line 707 of file GenesysParser.cpp.

7.58.4.97 symbol_name()

```
std::string yy::genesyspp_parser::symbol_name (
    symbol_kind_type yysymbol ) [static]
```

The user-facing name of the symbol whose (internal) number is YYSYMBOL. No bounds checking.

Definition at line 1965 of file GenesysParser.cpp.

Here is the caller graph for this function:

7.58.5 Member Data Documentation

7.58.5.1 YYNTOKENS

`const symbol_kind_type yy::genesyspp_parser::YYNTOKENS = symbol_kind::↩ YYNTOKENS [static]`

The number of tokens.

Definition at line 796 of file GenesysParser.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.cpp

7.59 GenesysQtGUI Class Reference

```
#include <GenesysQtGUI.h>
```

Inheritance diagram for GenesysQtGUI:

Collaboration diagram for GenesysQtGUI:

Public Member Functions

- GenesysQtGUI ()
- virtual int main (int argc, char ∗∗argv)

7.59.1 Detailed Description

Definition at line 22 of file GenesysQtGUI.h.

7.59.2 Constructor & Destructor Documentation

7.59.2.1 GenesysQtGUI() `GenesysQtGUI::GenesysQtGUI ( )`

Definition at line 18 of file GenesysQtGUI.cpp.

7.59.3 Member Function Documentation

7.59.3.1 main()

```
int GenesysQtGUI::main (
    int argc,
    char ** argv )  [virtual]
```

Implements GenesysApplication_if.

Definition at line 21 of file GenesysQtGUI.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/GenesysQtGUI.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/GenesysQtGUI.cpp

7.60 GenesysShell_if Class Reference

```
#include <GenesysShell_if.h>
```

Inheritance diagram for GenesysShell_if:

Collaboration diagram for GenesysShell_if:

Public Member Functions

- virtual void openModel (std::string filename)=0
- virtual void saveModelAs (std::string filename)=0
- virtual void saveModel ()=0
- virtual void listElements ()=0
- virtual void listComponents ()=0
- virtual void listHosts ()=0
- virtual void listPlugins ()=0
- virtual void deleteTraceFiles ()=0
- virtual void traceLevel (Util::TraceLevel tracelevel)=0
- virtual void addPlugin (std::string filename)=0
- virtual void addFromFile (std::string filename)=0
- virtual void readCommandsFromFile (std::string filename)=0
- virtual void redirectTrace (std::string trace, std::string dest, std::string filename)=0
- virtual void closeModel ()=0
- virtual void createModel ()=0
- virtual void execLinuxCommand (std::string command)=0
- virtual void verboseMode (bool on)=0
- virtual void check ()=0
- virtual void getGenesysInfo ()=0
- virtual void getCommandLine ()=0

- virtual void sendFile (std::string filename, std::string hostname, std::string portname)=0
- virtual void setActivationCode (std::string code)=0
- virtual void receiveFile (std::string filename)=0
- virtual void startSimulation ()=0
- virtual void stepSimulation ()=0
- virtual void stopSimulation ()=0
- virtual void showInit ()=0
- virtual void showHelp ()=0
- virtual void showHostName ()=0

7.60.1 Detailed Description

Definition at line 20 of file GenesysShell_if.h.

7.60.2 Member Function Documentation

7.60.2.1 addFromFile()

```
virtual void GenesysShell_if::addFromFile (
    std::string filename ) [pure virtual]
```

7.60.2.2 addPlugin()

```
virtual void GenesysShell_if::addPlugin (
    std::string filename ) [pure virtual]
```

7.60.2.3 check()

```
virtual void GenesysShell_if::check ( ) [pure virtual]
```

7.60.2.4 closeModel()

```
virtual void GenesysShell_if::closeModel ( ) [pure virtual]
```

7.60.2.5 createModel()

```
virtual void GenesysShell_if::createModel ( ) [pure virtual]
```

7.60.2.6 deleteTraceFiles()

```
virtual void GenesysShell_if::deleteTraceFiles ( ) [pure virtual]
```

7.60.2.7 execLinuxCommand()

`virtual void GenesysShell_if::execLinuxCommand ( std::string command ) [pure virtual]`

7.60.2.8 getCommandLine()

`virtual void GenesysShell_if::getCommandLine ( ) [pure virtual]`

7.60.2.9 getGenesysInfo()

`virtual void GenesysShell_if::getGenesysInfo ( ) [pure virtual]`

7.60.2.10 listComponents()

`virtual void GenesysShell_if::listComponents ( ) [pure virtual]`

7.60.2.11 listElements()

`virtual void GenesysShell_if::listElements ( ) [pure virtual]`

7.60.2.12 listHosts()

`virtual void GenesysShell_if::listHosts ( ) [pure virtual]`

7.60.2.13 listPlugins()

`virtual void GenesysShell_if::listPlugins ( ) [pure virtual]`

7.60.2.14 openModel()

`virtual void GenesysShell_if::openModel ( std::string filename ) [pure virtual]`

7.60.2.15 readCommandsFromFile()

`virtual void GenesysShell_if::readCommandsFromFile ( std::string filename ) [pure virtual]`

7.60.2.16 receiveFile()

`virtual void GenesysShell_if::receiveFile ( std::string filename ) [pure virtual]`

7.60.2.17 redirectTrace()

```
virtual void GenesysShell_if::redirectTrace (
    std::string trace,
    std::string dest,
    std::string filename ) [pure virtual]
```

7.60.2.18 saveModel()

```
virtual void GenesysShell_if::saveModel ( ) [pure virtual]
```

7.60.2.19 saveModelAs()

```
virtual void GenesysShell_if::saveModelAs (
    std::string filename ) [pure virtual]
```

7.60.2.20 sendFile()

```
virtual void GenesysShell_if::sendFile (
    std::string filename,
    std::string hostname,
    std::string portname ) [pure virtual]
```

7.60.2.21 setActivationCode()

```
virtual void GenesysShell_if::setActivationCode (
    std::string code ) [pure virtual]
```

7.60.2.22 showHelp()

```
virtual void GenesysShell_if::showHelp ( ) [pure virtual]
```

7.60.2.23 showHostName()

```
virtual void GenesysShell_if::showHostName ( ) [pure virtual]
```

7.60.2.24 showInit()

```
virtual void GenesysShell_if::showInit ( ) [pure virtual]
```

7.60.2.25 startSimulation()

```
virtual void GenesysShell_if::startSimulation ( ) [pure virtual]
```

7.60.2.26 stepSimulation()

```
virtual void GenesysShell_if::stepSimulation ( ) [pure virtual]
```

7.60.2.27 stopSimulation()

`virtual void GenesysShell_if::stopSimulation ( ) [pure virtual]`

7.60.2.28 traceLevel()

`virtual void GenesysShell_if::traceLevel ( Util::TraceLevel tracelevel ) [pure virtual]`

7.60.2.29 verboseMode()

`virtual void GenesysShell_if::verboseMode ( bool on ) [pure virtual]`

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/terminal/GenesysShell_if.h

7.61 GenesysTerminalApp Class Reference

```
#include <GenesysTerminalApp.h>
```

Inheritance diagram for GenesysTerminalApp:

Collaboration diagram for GenesysTerminalApp:

Public Member Functions

- GenesysTerminalApp ()
- virtual ~GenesysTerminalApp ()=default
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.61.1 Detailed Description

Definition at line 21 of file GenesysTerminalApp.h.

7.61.2 Constructor & Destructor Documentation

7.61.2.1 GenesysTerminalApp() `GenesysTerminalApp::GenesysTerminalApp ( )`

Definition at line 22 of file GenesysTerminalApp.cpp.

7.61.2.2 ~GenesysTerminalApp() `virtual GenesysTerminalApp::~GenesysTerminalApp ( ) [virtual], [default]`

7.61.3 Member Function Documentation

7.61.3.1 main() `int GenesysTerminalApp::main ( int argc, char ** argv ) [virtual]`

Implements BaseConsoleGenesysApplication.

Definition at line 102 of file GenesysTerminalApp.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/terminal/GenesysTerminalApp.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/terminal/GenesysTerminalApp.cpp

7.62 Hold Class Reference

```
#include <Hold.h>
```

Inheritance diagram for Hold:

Collaboration diagram for Hold:

Public Member Functions

- Hold (Model ∗model, std::string name="")
- virtual ∼Hold ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.62.1 Detailed Description

Hold module DESCRIPTION This module will hold an entity in a queue to wait for a signal, wait for a specified condition to become true (scan), or be held infinitely (to be removed later with the Remove module). If the entity is holding for a signal, the Signal module is used elsewhere in the model to allow the entity to move on to the next module. If the entity is holding for a given condition to be true, the entity will remain at the module (either in a defined or internal queue) until the condition(s) becomes true. When the entity is in an infinite hold, the Remove module is used elsewhere in the model to allow the entity to continue processing. TYPICAL USES Waiting for a traffic light to turn green Holding a part for authorization Checking the status of a machine or operator to continue a process PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Indicates the reasoning for holding the entity within a specified or internal queue. Wait for Signal will hold the entity until a signal of the same value is received. Scan for Condition will hold the entity until the specified condition becomes true. Infinite Hold will hold the entity until it is removed from the queue by a Remove module. Wait for Value Signal code for the waiting entity. Applies only when Type is Wait for Signal. Limit Maximum number of waiting entities that will be released upon receipt of a signal. Applies only when Type is Wait for Signal. Condition Specifies the condition that will be evaluated to hold the entity at the module. If the condition is evaluated to true, the entity leaves the module immediately. If the condition is false, the entity will wait in the associated queue until the condition becomes true. Applies only when Type is Scan for Condition. Queue Type Determines the type of queue used to hold the entities. If Queue is selected, the queue name is specified. If Set is selected, the queue set and member in the set are specified. If Internal is selected, an internal queue is used to hold all waiting entities. Attribute and Expression are additional methods for defining the queue to be used. Queue Name This field is visible only if Queue Type is Queue, and it defines the symbol name of the queue. Set Name This field is visible only if Queue Type is Set, and it defines the queue set that contains the queue being referenced. Set Index This field is visible only if Queue Type is Set, and it defines the index into the queue set. Note that this is the index into the set and not the name of the queue in the set. For example, the only valid entry for a queue set containing three members is an expression that evaluates to 1, 2, or 3. Attribute This field is visible only if Queue Type is Attribute. The attribute entered in this field will be evaluated to indicate which queue is to be used. Expression This field is visible only if Queue Type is Expression. The expression entered in this field will be evaluated to indicate which queue is to be used.

Definition at line 75 of file Hold.h.

7.62.2 Constructor & Destructor Documentation

7.62.2.1 Hold()

```
Hold::Hold (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Hold.cpp.

Here is the caller graph for this function:

7.62.2.2 ~Hold() `virtual Hold::~Hold ( ) [virtual], [default]`

7.62.3 Member Function Documentation

7.62.3.1 _check() `bool Hold::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 56 of file Hold.cpp.

7.62.3.2 _execute() `void Hold::_execute ( Entity * entity ) [protected], [virtual]`

Implements ModelComponent.

Definition at line 34 of file Hold.cpp.

Here is the call graph for this function:

7.62.3.3 _initBetweenReplications() `void Hold::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 47 of file Hold.cpp.

7.62.3.4 _loadInstance()

```
bool Hold::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 39 of file Hold.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.62.3.5 _saveInstance()

```
std::map< std::string, std::string > * Hold::_saveInstance ( ) [protected],
[virtual]
```

Reimplemented from ModelComponent.

Definition at line 50 of file Hold.cpp.

Here is the call graph for this function:

7.62.3.6 GetPluginInformation() `PluginInformation * Hold::GetPluginInformation ( ) [static]`

Definition at line 63 of file Hold.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.62.3.7 LoadInstance() `ModelComponent * Hold::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 24 of file Hold.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.62.3.8 show() `std::string Hold::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 20 of file Hold.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Hold.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Hold.cpp

7.63 HypothesisTester_if Class Reference

```
#include <HypothesisTester_if.h>
```

Inheritance diagram for HypothesisTester_if:

Collaboration diagram for HypothesisTester_if:

Classes

- class ConfidenceInterval
- class TestResult

Public Types

- enum H1Comparition { DIFFERENT = 1 , LESS_THAN = 2 , GREATER_THAN = 3 }

Public Member Functions

- virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval (double avg, double stddev, unsigned int n, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval (double prop, unsigned int n, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval (double var, unsigned int n, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval averageDifferenceConfidenceInterval (double avg1, double stddev1, unsigned int n1, double avg2, double stddev2, unsigned int n2, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval varianceRatioConfidenceInterval (double var1, unsigned int n1, double var2, unsigned int n2, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval (std::string sampleDataFilename, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval (std::string sampleData↩Filename, checkProportionFunction function, double confidenceLevel)=0
- virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval (std::string sampleDataFilename, double confidenceLevel)=0

- virtual HypothesisTester_if::TestResult testAverage (double avg, double stddev, unsigned int n, double avg↩Sample, double confidenceLevel, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testProportion (double prop, unsigned int n, double proptest, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testVariance (double var, unsigned int n, double vartest, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testAverage (double avg1, unsigned int n1, double avg2, unsigned int n2, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testProportion (double prop1, unsigned int n1, double prop2, unsigned int n2, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testVariance (double var1, unsigned int n1, double var2, unsigned int n2, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testAverage (std::string sampleDataFilename, double avgSample, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testProportion (std::string sampleDataFilename, checkProportionFunction function, double proptest, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testVariance (std::string sampleDataFilename, double vartest, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testAverage (std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testProportion (std::string firstSampleDataFilename, std::string secondSampleDataFilename, checkProportionFunction function, HypothesisTester_if::H1Comparition comp)=0
- virtual HypothesisTester_if::TestResult testVariance (std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp)=0

7.63.1 Detailed Description

Interface for parametric hypothesis tests based on a datafile or parameters. All tests are suposed to be based on samples with unknown population parameters

Definition at line 25 of file HypothesisTester_if.h.

7.63.2 Member Enumeration Documentation

7.63.2.1 H1Comparition `enum HypothesisTester_if::H1Comparition`

Enumerator

DIFFERENT		
LESS_THAN		
GREATER_THAN		

Definition at line 45 of file HypothesisTester_if.h.

7.63.3 Member Function Documentation

7.63.3.1 averageConfidenceInterval() [1/2]

```
virtual HypothesisTester_if::ConfidenceInterval HypothesisTester_if::averageConfidenceInterval (
            double avg,
            double stddev,
            unsigned int n,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

Here is the caller graph for this function:

7.63.3.2 averageConfidenceInterval() [2/2]

```
virtual HypothesisTester_if::ConfidenceInterval HypothesisTester_if::averageConfidenceInterval (
            std::string sampleDataFilename,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.3 averageDifferenceConfidenceInterval()

```
virtual HypothesisTester_if::ConfidenceInterval HypothesisTester_if::averageDifferenceConfidenceInterval (
            double avg1,
            double stddev1,
            unsigned int n1,
            double avg2,
            double stddev2,
            unsigned int n2,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.4 proportionConfidenceInterval() [1/2]

```
virtual HypothesisTester_if::ConfidenceInterval HypothesisTester_if::proportionConfidenceInterval (
            double prop,
            unsigned int n,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.5 proportionConfidenceInterval() [2/2]

```
virtual HypothesisTester_if::ConfidenceInterval
HypothesisTester_if::proportionConfidenceInterval (
            std::string sampleDataFilename,
            checkProportionFunction function,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.6 testAverage() [1/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::test↩
Average (
            double avg,
            double stddev,
            unsigned int n,
            double avgSample,
            double confidenceLevel,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

Here is the caller graph for this function:

7.63.3.7 testAverage() [2/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::test↩
Average (
            double avg1,
            unsigned int n1,
            double avg2,
            unsigned int n2,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.8 testAverage() [3/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::test↩
Average (
            std::string firstSampleDataFilename,
            std::string secondSampleDataFilename,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.9 testAverage() [4/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::testAverage (
            std::string sampleDataFilename,
            double avgSample,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.10 testProportion() [1/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::testProportion (
            double prop,
            unsigned int n,
            double proptest,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.11 testProportion() [2/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::testProportion (
            double prop1,
            unsigned int n1,
            double prop2,
            unsigned int n2,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.12 testProportion() [3/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::testProportion (
            std::string firstSampleDataFilename,
            std::string secondSampleDataFilename,
            checkProportionFunction function,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.13 testProportion() [4/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if::testProportion (
            std::string sampleDataFilename,
            checkProportionFunction function,
            double proptest,
            HypothesisTester_if::H1Comparition comp ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.14 testVariance() [1/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if↩
::testVariance (
            double var,
            unsigned int n,
            double vartest,
            HypothesisTester_if::H1Comparition comp )  [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.15 testVariance() [2/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if↩
::testVariance (
            double var1,
            unsigned int n1,
            double var2,
            unsigned int n2,
            HypothesisTester_if::H1Comparition comp )  [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.16 testVariance() [3/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if↩
::testVariance (
            std::string firstSampleDataFilename,
            std::string secondSampleDataFilename,
            HypothesisTester_if::H1Comparition comp )  [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.17 testVariance() [4/4]

```
virtual HypothesisTester_if::TestResult HypothesisTester_if↩
::testVariance (
            std::string sampleDataFilename,
            double vartest,
            HypothesisTester_if::H1Comparition comp )  [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.18 varianceConfidenceInterval() [1/2]

```
virtual HypothesisTester_if::ConfidenceInterval
HypothesisTester_if::varianceConfidenceInterval (
            double var,
            unsigned int n,
            double confidenceLevel )  [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.19 varianceConfidenceInterval() [2/2]

```
virtual HypothesisTester_if::ConfidenceInterval
HypothesisTester_if::varianceConfidenceInterval (
            std::string sampleDataFilename,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

7.63.3.20 varianceRatioConfidenceInterval()

```
virtual HypothesisTester_if::ConfidenceInterval
HypothesisTester_if::varianceRatioConfidenceInterval (
            double var1,
            unsigned int n1,
            double var2,
            unsigned int n2,
            double confidenceLevel ) [pure virtual]
```

Implemented in HypothesisTesterDefaultImpl1.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/HypothesisTester_if.h

7.64 HypothesisTesterDefaultImpl1 Class Reference

```
#include <HypothesisTesterDefaultImpl1.h>
```

Inheritance diagram for HypothesisTesterDefaultImpl1:

Collaboration diagram for HypothesisTesterDefaultImpl1:

Public Member Functions

- HypothesisTesterDefaultImpl1 ()
- virtual ~HypothesisTesterDefaultImpl1 ()=default
- virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval (double avg, double stddev, unsigned int n, double confidenceLevel)
- virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval (double prop, unsigned int n, double confidenceLevel)
- virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval (double var, unsigned int n, double confidenceLevel)
- virtual HypothesisTester_if::ConfidenceInterval averageDifferenceConfidenceInterval (double avg1, double stddev1, unsigned int n1, double avg2, double stddev2, unsigned int n2, double confidenceLevel)

- virtual HypothesisTester_if::ConfidenceInterval varianceRatioConfidenceInterval (double var1, unsigned int n1, double var2, unsigned int n2, double confidenceLevel)
- virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval (std::string sampleDataFilename, double confidenceLevel)
- virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval (std::string sampleData↩ Filename, checkProportionFunction function, double confidenceLevel)
- virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval (std::string sampleDataFilename, double confidenceLevel)
- virtual HypothesisTester_if::TestResult testAverage (double avg, double stddev, unsigned int n, double avg↩ Sample, double confidenceLevel, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testProportion (double prop, unsigned int n, double proptest, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testVariance (double var, unsigned int n, double vartest, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testAverage (double avg1, unsigned int n1, double avg2, unsigned int n2, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testProportion (double prop1, unsigned int n1, double prop2, unsigned int n2, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testVariance (double var1, unsigned int n1, double var2, unsigned int n2, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testAverage (std::string sampleDataFilename, double avgSample, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testProportion (std::string sampleDataFilename, checkProportionFunction function, double proptest, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testVariance (std::string sampleDataFilename, double vartest, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testAverage (std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testProportion (std::string firstSampleDataFilename, std::string secondSampleDataFilename, checkProportionFunction function, HypothesisTester_if::H1Comparition comp)
- virtual HypothesisTester_if::TestResult testVariance (std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp)

Additional Inherited Members

7.64.1 Detailed Description

Definition at line 18 of file HypothesisTesterDefaultImpl1.h.

7.64.2 Constructor & Destructor Documentation

7.64.2.1 HypothesisTesterDefaultImpl1() `HypothesisTesterDefaultImpl1::HypothesisTesterDefault↩ Impl1 ( )`

Definition at line 19 of file HypothesisTesterDefaultImpl1.cpp.

7.64.2.2 ~HypothesisTesterDefaultImpl1()

```
virtual HypothesisTesterDefaultImpl1::~HypothesisTesterDefaultImpl1 ( ) [virtual], [default]
```

7.64.3 Member Function Documentation

7.64.3.1 averageConfidenceInterval() [1/2]

```
HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::averageConfidenceInterval (
            double avg,
            double stddev,
            unsigned int n,
            double confidenceLevel ) [virtual]
```

Implements HypothesisTester_if.

Definition at line 24 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.64.3.2 averageConfidenceInterval() [2/2]

```
HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::averageConfidenceInterval (
            std::string sampleDataFilename,
            double confidenceLevel ) [virtual]
```

Implements HypothesisTester_if.

Definition at line 68 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

7.64.3.3 averageDifferenceConfidenceInterval()

```
HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::averageDifferenceConfidenceInterval (
            double avg1,
            double stddev1,
            unsigned int n1,
            double avg2,
            double stddev2,
            unsigned int n2,
            double confidenceLevel )  [virtual]
```

Implements HypothesisTester_if.

Definition at line 42 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

7.64.3.4 proportionConfidenceInterval() [1/2]

```
HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::proportionConfidenceInterval (
            double prop,
            unsigned int n,
            double confidenceLevel )  [virtual]
```

Implements HypothesisTester_if.

Definition at line 30 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.64.3.5 proportionConfidenceInterval() [2/2]

```
HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::proportionConfidenceInterval (
            std::string sampleDataFilename,
            checkProportionFunction function,
            double confidenceLevel ) [virtual]
```

Implements HypothesisTester_if.

Definition at line 74 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

7.64.3.6 testAverage() [1/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testAverage (
            double avg,
            double stddev,
            unsigned int n,
            double avgSample,
            double confidenceLevel,
            HypothesisTester_if::H1Comparition comp )  [virtual]
```

Implements HypothesisTester_if.

Definition at line 96 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

7.64.3.7 testAverage() [2/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testAverage (
            double avg1,
            unsigned int n1,
            double avg2,
            unsigned int n2,
            HypothesisTester_if::H1Comparition comp )  [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 121 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.8 testAverage() [3/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testAverage (
            std::string firstSampleDataFilename,
            std::string secondSampleDataFilename,
            HypothesisTester_if::H1Comparition comp )  [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 147 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.9 testAverage() [4/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testAverage (
            std::string sampleDataFilename,
            double avgSample,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 134 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.10 testProportion() [1/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testProportion (
            double prop,
            unsigned int n,
            double proptest,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 112 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.11 testProportion() [2/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testProportion (
            double prop1,
            unsigned int n1,
            double prop2,
            unsigned int n2,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 125 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.12 testProportion() [3/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testProportion (
            std::string firstSampleDataFilename,
            std::string secondSampleDataFilename,
            checkProportionFunction function,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 151 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.13 testProportion() [4/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testProportion (
            std::string sampleDataFilename,
            checkProportionFunction function,
            double proptest,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 138 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.14 testVariance() [1/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testVariance (
            double var,
            unsigned int n,
            double vartest,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 116 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.15 testVariance() [2/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testVariance (
            double var1,
            unsigned int n1,
            double var2,
            unsigned int n2,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 129 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.16 testVariance() [3/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testVariance (
            std::string firstSampleDataFilename,
            std::string secondSampleDataFilename,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 155 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.17 testVariance() [4/4]

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1↩
::testVariance (
            std::string sampleDataFilename,
            double vartest,
            HypothesisTester_if::H1Comparition comp ) [virtual]
```

TODO: not implemented yet

Implements HypothesisTester_if.

Definition at line 142 of file HypothesisTesterDefaultImpl1.cpp.

7.64.3.18 varianceConfidenceInterval() [1/2]

```
HypothesisTester_if::ConfidenceInterval Hypothesis↩
TesterDefaultImpl1::varianceConfidenceInterval (
            double var,
            unsigned int n,
            double confidenceLevel ) [virtual]
```

Implements HypothesisTester_if.

Definition at line 36 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.64.3.19 varianceConfidenceInterval() [2/2] HypothesisTester_if::ConfidenceInterval Hypothesis↩ TesterDefaultImpl1::varianceConfidenceInterval (

```
std::string sampleDataFilename,
double confidenceLevel ) [virtual]
```

Implements HypothesisTester_if.

Definition at line 88 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

7.64.3.20 varianceRatioConfidenceInterval() HypothesisTester_if::ConfidenceInterval Hypothesis↩ TesterDefaultImpl1::varianceRatioConfidenceInterval (

```
double var1,
unsigned int n1,
double var2,
unsigned int n2,
double confidenceLevel ) [virtual]
```

Implements HypothesisTester_if.

Definition at line 56 of file HypothesisTesterDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/HypothesisTesterDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/HypothesisTesterDefaultImpl1.cpp

7.65 Leave Class Reference

```
#include <Leave.h>
```

Inheritance diagram for Leave:

Collaboration diagram for Leave:

Public Member Functions

- Leave (Model ∗model, std::string name="")
- virtual ∼Leave ()=default
- virtual std::string show ()
- void setStation (Station ∗_station)
- Station ∗ getStation () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.65.1 Detailed Description

Leave module DESCRIPTION The Leave module is used to transfer an entity to a station or module. An entity may be transferred in two ways. It can be transferred to a module that defines a station by referencing the station and routing, conveying, or transporting to that station, or a graphical connection can be used to transfer an entity to another module. When an entity arrives at a Leave module, it may wait to obtain a transfer device (resource, transporter, or conveyor). When the transfer device has been obtained, the entity may experience a loading delay. Finally, the entity is transferred from this module to a destination module or station. TYPICAL USES The end of a part's production in a series of parallel processes where the part needs a forklift to be transferred to shipping PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Allocation Type of category to which the entity's incurred delay time and cost will be added. Delay Specifies a load time incurred after getting a transfer device. Units Time units used for the delay time. Transfer Out Determines whether a resource (Seize Resource), transporter (Request Transporter), or conveyor (Access Conveyor) is required prior to transferring the entity out of this module. Priority Indicates the priority of the module when either seizing a resource or requesting a transporter when there are entities waiting for that resource/transporter from other modules. This field is not visible when the Transfer Type is None or Access Conveyor. Transporter Name Name of the transporter to request. Queue Type Type of queue, either a single Queue, queue Set, Internal queue, Attribute, or Expression. Queue Name Name of the individual queue. Queue Set Name Name of the queue set. Set Index Defines the index into the queue set. Note that this is the index into the set and not the name of the queue in the set. Queue Attribute Name The attribute name that will be evaluated to indicate which queue is to be used. Queue Expression The expression that will be evaluated to indicate which queue is to be used. Selection Rule Method of selecting among available transporters in a set. Cyclical will cycle through available members. Random will randomly select a member. Preferred Order will always select the first available member. Specific Member requires an input attribute value to specify which member of the set (previously saved in the Save Attribute field). Largest Distance selects the transporter farthest away, and Smallest Distance selects the closest transporter. Save Attribute Attribute name used to store the index number into the set of the member that is chosen. This attribute can later be referenced with the Specific Member selection rule. Active when Transfer Out is Request Transporter. Index Set Attribute name whose value identifies the index number into the set of the member requested. The entity must have a value for the attribute before utilizing this option. Resource Type Type of resource for seizing, either specifying a particular Resource, selecting from a pool of resources (that is, a resource Set), Attribute, or Expression. Resource Name Name of the resource to seize. Conveyor Name Name of the conveyor to access.

of Cells Number of contiguous cells the entity requires.

Connect Type Determines if the entity is to Route, Convey, or Transport to another station or Connect to another module. Move Time Time to route from this module to the destination station. Units Time units used for the move time. Station Type The entity's destination station type either an individual Station, a station based on an Attribute or Expression value, or By Sequence. Station Name Name of the individual destination station. Attribute Name The attribute name that will be evaluated to indicate the station. Expression The expression that will be evaluated to indicate the station.

Definition at line 94 of file Leave.h.

7.65.2 Constructor & Destructor Documentation

7.65.2.1 Leave()

```
Leave::Leave (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Leave.cpp.

Here is the caller graph for this function:

7.65.2.2 ~Leave()

```
virtual Leave::~Leave ( ) [virtual], [default]
```

7.65.3 Member Function Documentation

7.65.3.1 _check()

```
bool Leave::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 69 of file Leave.cpp.

Here is the call graph for this function:

7.65.3.2 _createInternalElements() `void Leave::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 81 of file Leave.cpp.

Here is the call graph for this function:

7.65.3.3 _execute() `void Leave::_execute ( Entity * entity ) [protected], [virtual]`

Implements ModelComponent.

Definition at line 42 of file Leave.cpp.

Here is the call graph for this function:

7.65.3.4 _initBetweenReplications() `void Leave::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 59 of file Leave.cpp.

Here is the call graph for this function:

7.65.3.5 _loadInstance() `bool Leave::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 49 of file Leave.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.65.3.6 _saveInstance() `std::map< std::string, std::string > * Leave::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 63 of file Leave.cpp.

Here is the call graph for this function:

7.65.3.7 GetPluginInformation() `PluginInformation * Leave::GetPluginInformation ( ) [static]`

Definition at line 75 of file Leave.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.65.3.8 getStation() `Station * Leave::getStation ( ) const`

Definition at line 38 of file Leave.cpp.

7.65.3.9 LoadInstance()

```
ModelComponent * Leave::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 24 of file Leave.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.65.3.10 setStation()

```
void Leave::setStation (
    Station * _station )
```

Definition at line 34 of file Leave.cpp.

Here is the caller graph for this function:

7.65.3.11 show() `std::string Leave::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 20 of file Leave.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Leave.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Leave.cpp

7.66 LicenceManager Class Reference

```
#include <LicenceManager.h>
```

Collaboration diagram for LicenceManager:

Public Member Functions

- LicenceManager (Simulator *simulator)
- virtual ~LicenceManager ()=default
- const std::string showLicence () const
- const std::string showLimits () const
- const std::string showActivationCode () const
- bool lookforActivationCode ()
- bool insertActivationCode ()
- void removeActivationCode ()
- unsigned int getModelComponentsLimit ()
- unsigned int getModelElementsLimit ()
- unsigned int getEntityLimit ()
- unsigned int getHostsLimit ()
- unsigned int getThreadsLimit ()

7.66.1 Detailed Description

LicenceManager just prints a licence agreement and checks for some resource limits.

Definition at line 25 of file LicenceManager.h.

7.66.2 Constructor & Destructor Documentation

7.66.2.1 LicenceManager() `LicenceManager::LicenceManager ( Simulator * simulator )`

Definition at line 18 of file LicenceManager.cpp.

7.66.2.2 ~LicenceManager() `virtual LicenceManager::~LicenceManager ( ) [virtual], [default]`

7.66.3 Member Function Documentation

7.66.3.1 getEntityLimit() `unsigned int LicenceManager::getEntityLimit ( )`

Definition at line 74 of file LicenceManager.cpp.

7.66.3.2 getHostsLimit() `unsigned int LicenceManager::getHostsLimit ( )`

Definition at line 78 of file LicenceManager.cpp.

7.66.3.3 getModelComponentsLimit() `unsigned int LicenceManager::getModelComponentsLimit ( )`

Definition at line 66 of file LicenceManager.cpp.

Here is the caller graph for this function:

7.66.3.4 getModelElementsLimit() `unsigned int LicenceManager::getModelElementsLimit ( )`

Definition at line 70 of file LicenceManager.cpp.

Here is the caller graph for this function:

7.66.3.5 getThreadsLimit() `unsigned int LicenceManager::getThreadsLimit ( )`

Definition at line 82 of file LicenceManager.cpp.

7.66.3.6 insertActivationCode() `bool LicenceManager::insertActivationCode ( )`

Definition at line 57 of file LicenceManager.cpp.

7.66.3.7 lookforActivationCode() `bool LicenceManager::lookforActivationCode ( )`

Definition at line 52 of file LicenceManager.cpp.

7.66.3.8 removeActivationCode() `void LicenceManager::removeActivationCode ( )`

Definition at line 62 of file LicenceManager.cpp.

7.66.3.9 showActivationCode() `const std::string LicenceManager::showActivationCode ( ) const`

Definition at line 47 of file LicenceManager.cpp.

7.66.3.10 showLicence() `const std::string LicenceManager::showLicence ( ) const`

Definition at line 33 of file LicenceManager.cpp.

Here is the caller graph for this function:

7.66.3.11 showLimits() `const std::string LicenceManager::showLimits ( ) const`

Definition at line 37 of file LicenceManager.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/LicenceManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/LicenceManager.cpp

7.67 List< T > Class Template Reference

```
#include <List.h>
```

Collaboration diagram for List< T >:

Public Types

- using CompFunct = std::function< bool(const T, const T) >

Public Member Functions

- List ()
- virtual ~List ()=default
- unsigned int size ()
- bool empty ()
- void clear ()
- void pop_front ()
- template<class Compare >
  void sort (Compare comp)
- std::list< T > * list () const
- T create ()
- template<typename U >
  T create (U arg)
- std::string show ()
- std::list< T >::iterator find (T element)
- void insert (T element)
- void remove (T element)
- void setAtRank (unsigned int rank, T element)
- T getAtRank (unsigned int rank)

- T next ()
- T front ()
- T last ()
- T previous ()
- T current ()
- void setSortFunc (CompFunct _sortFunc)

7.67.1 Detailed Description

template<typename T>
class List< T >

List corresponds to an extended version of the list that must guarantee the consistency of the elements that make up the simulation model.

Definition at line 32 of file List.h.

7.67.2 Member Typedef Documentation

7.67.2.1 CompFunct `template<typename T >`

```
using List< T >::CompFunct = std::function<bool(const T, const T) >
```

Definition at line 34 of file List.h.

7.67.3 Constructor & Destructor Documentation

7.67.3.1 List() `template<typename T >`

```
List< T >::List
```

Definition at line 73 of file List.h.

7.67.3.2 ~List() `template<typename T >`

```
virtual List< T >::~List ( )  [virtual], [default]
```

7.67.4 Member Function Documentation

7.67.4.1 clear() template<typename T >

```
void List< T >::clear
```

Definition at line 132 of file List.h.

Here is the caller graph for this function:

7.67.4.2 create() [1/2] template<typename T >

```
T List< T >::create
```

Definition at line 128 of file List.h.

7.67.4.3 create() [2/2] template<typename T >

```
template<typename U >
T List< T >::create (
            U arg )
```

Definition at line 228 of file List.h.

7.67.4.4 current() template<typename T >

```
T List< T >::current
```

Definition at line 218 of file List.h.

Here is the caller graph for this function:

7.67.4.5 empty() template<typename T >

bool List< T >::empty

Definition at line 109 of file List.h.

7.67.4.6 find() template<typename T >

std::list< T >::iterator List< T >::find (
T *element*)

Definition at line 173 of file List.h.

Here is the caller graph for this function:

7.67.4.7 front() template<typename T >

T List< T >::front

Definition at line 197 of file List.h.

Here is the caller graph for this function:

7.67.4.8 getAtRank()

```
template<typename T >
T List< T >::getAtRank (
            unsigned int rank )
```

Definition at line 136 of file List.h.

Here is the caller graph for this function:

7.67.4.9 insert()

```
template<typename T >
void List< T >::insert (
            T element )
```

Definition at line 105 of file List.h.

7.67.4.10 last() `template<typename T >`
`T List< T >::last`

Definition at line 205 of file List.h.

Here is the caller graph for this function:

7.67.4.11 list() `template<typename T >`
`std::list< T > * List< T >::list`

Definition at line 79 of file List.h.

7.67.4.12 next() `template<typename T >`
`T List< T >::next`

Definition at line 164 of file List.h.

Here is the caller graph for this function:

7.67.4.13 pop_front() template<typename T >

```
void List< T >::pop_front
```

Definition at line 113 of file List.h.

Here is the caller graph for this function:

7.67.4.14 previous() template<typename T >

```
T List< T >::previous
```

Definition at line 213 of file List.h.

7.67.4.15 remove() template<typename T >

```
void List< T >::remove (
            T element )
```

Definition at line 121 of file List.h.

Here is the caller graph for this function:

7.67.4.16 setAtRank() template<typename T >

```
void List< T >::setAtRank (
            unsigned int rank,
            T element )
```

Definition at line 148 of file List.h.

Here is the caller graph for this function:

7.67.4.17 setSortFunc() template<typename T >

```
void List< T >::setSortFunc (
            CompFunct _sortFunc )
```

Definition at line 223 of file List.h.

Here is the caller graph for this function:

7.67.4.18 show()

```
template<typename T >
std::string List< T >::show
```

Definition at line 95 of file List.h.

Here is the caller graph for this function:

7.67.4.19 size()

```
template<typename T >
unsigned int List< T >::size
```

Definition at line 83 of file List.h.

7.67.4.20 sort()

```
template<typename T >
template<class Compare >
void List< T >::sort (
            Compare comp )
```

Definition at line 233 of file List.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/util/List.h

7.68 yy::location Class Reference

Two points in a source file.

```
#include <location.hh>
```

Collaboration diagram for yy::location:

Public Types

- typedef position::filename_type filename_type

  Type for file name.
- typedef position::counter_type counter_type

  Type for line and column numbers.

Public Member Functions

- location (const position &b, const position &e)

  Construct a location from b to e.
- location (const position &p=position())

  Construct a 0-width location in p.
- location (filename_type *f, counter_type l=1, counter_type c=1)

  Construct a 0-width location in f, l, c.
- void initialize (filename_type *f=YY_NULLPTR, counter_type l=1, counter_type c=1)

  Initialization.

Line and Column related manipulators

- void step ()

  Reset initial location to final location.
- void columns (counter_type count=1)

  Extend the current location to the COUNT next columns.
- void lines (counter_type count=1)

  Extend the current location to the COUNT next lines.

Public Attributes

- position begin

  Beginning of the located region.
- position end

  End of the located region.

7.68.1 Detailed Description

Two points in a source file.

Definition at line 165 of file location.hh.

7.68.2 Member Typedef Documentation

7.68.2.1 counter_type `typedef position::counter_type yy::location::counter_type`

Type for line and column numbers.

Definition at line 171 of file location.hh.

7.68.2.2 filename_type `typedef position::filename_type yy::location::filename_type`

Type for file name.

Definition at line 169 of file location.hh.

7.68.3 Constructor & Destructor Documentation

7.68.3.1 location() [1/3]

```
yy::location::location (
    const position & b,
    const position & e )  [inline]
```

Construct a location from *b* to *e*.

Definition at line 174 of file location.hh.

7.68.3.2 location() [2/3]

```
yy::location::location (
    const position & p = position () )  [inline], [explicit]
```

Construct a 0-width location in *p*.

Definition at line 180 of file location.hh.

7.68.3.3 location() [3/3]

```
yy::location::location (
    filename_type * f,
    counter_type l = 1,
    counter_type c = 1 )  [inline], [explicit]
```

Construct a 0-width location in *f*, *l*, *c*.

Definition at line 186 of file location.hh.

7.68.4 Member Function Documentation

7.68.4.1 columns()

```
void yy::location::columns (
    counter_type count = 1 ) [inline]
```

Extend the current location to the COUNT next columns.

Definition at line 213 of file location.hh.

Here is the caller graph for this function:

7.68.4.2 initialize()

```
void yy::location::initialize (
    filename_type * f = YY_NULLPTR,
    counter_type l = 1,
    counter_type c = 1 ) [inline]
```

Initialization.

Definition at line 195 of file location.hh.

Here is the call graph for this function:

7.68.4.3 lines()

```
void yy::location::lines (
    counter_type count = 1 ) [inline]
```

Extend the current location to the COUNT next lines.

Definition at line 219 of file location.hh.

Here is the call graph for this function:

7.68.4.4 step() `void yy::location::step ( ) [inline]`

Reset initial location to final location.

Definition at line 207 of file location.hh.

7.68.5 Member Data Documentation

7.68.5.1 begin `position yy::location::begin`

Beginning of the located region.

Definition at line 228 of file location.hh.

7.68.5.2 end `position yy::location::end`

End of the located region.

Definition at line 230 of file location.hh.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/location.hh

7.69 LSODE Class Reference

```
#include <LSODE.h>
```

Inheritance diagram for LSODE:

Collaboration diagram for LSODE:

Public Member Functions

- LSODE (Model ∗model, std::string name="")
- virtual ∼LSODE ()=default
- virtual std::string show ()
- void setDiffEquations (Formula ∗formula)
- Formula ∗ getDiffEquations () const
- void setTimeVariable (Variable ∗_timeVariable)
- Variable ∗ getTimeVariable () const
- void setStep (double _step)
- double getStep () const
- void setVariables (Variable ∗_variables)
- Variable ∗ getVariables () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.69.1 Detailed Description

This component ...

Definition at line 24 of file LSODE.h.

7.69.2 Constructor & Destructor Documentation

7.69.2.1 LSODE()

```
LSODE::LSODE (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file LSODE.cpp.

Here is the caller graph for this function:

7.69.2.2 ~LSODE()

```
virtual LSODE::~LSODE ( ) [virtual], [default]
```

7.69.3 Member Function Documentation

7.69.3.1 _check()

```
bool LSODE::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 149 of file LSODE.cpp.

7.69.3.2 _execute()

```
void LSODE::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 116 of file LSODE.cpp.

Here is the call graph for this function:

7.69.3.3 _initBetweenReplications()

```
void LSODE::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 140 of file LSODE.cpp.

7.69.3.4 _loadInstance()

```
bool LSODE::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 132 of file LSODE.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.69.3.5 _saveInstance() `std::map< std::string, std::string > * LSODE::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 143 of file LSODE.cpp.

Here is the call graph for this function:

7.69.3.6 getDiffEquations() `Formula * LSODE::getDiffEquations ( ) const`

Definition at line 38 of file LSODE.cpp.

7.69.3.7 GetPluginInformation() `PluginInformation * LSODE::GetPluginInformation ( ) [static]`

Definition at line 156 of file LSODE.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.69.3.8 getStep() `double LSODE::getStep ( ) const`

Definition at line 54 of file LSODE.cpp.

7.69.3.9 getTimeVariable() `Variable * LSODE::getTimeVariable ( ) const`

Definition at line 46 of file LSODE.cpp.

7.69.3.10 getVariables() `Variable * LSODE::getVariables ( ) const`

Definition at line 62 of file LSODE.cpp.

7.69.3.11 LoadInstance()

```
ModelComponent * LSODE::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields )  [static]
```

Definition at line 24 of file LSODE.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.69.3.12 setDiffEquations()

```
void LSODE::setDiffEquations (
    Formula * formula )
```

Definition at line 34 of file LSODE.cpp.

7.69.3.13 setStep()

```
void LSODE::setStep (
    double _step )
```

Definition at line 50 of file LSODE.cpp.

7.69.3.14 setTimeVariable()

```
void LSODE::setTimeVariable (
    Variable * _timeVariable )
```

Definition at line 42 of file LSODE.cpp.

7.69.3.15 setVariables()

```
void LSODE::setVariables (
    Variable * _variables )
```

Definition at line 58 of file LSODE.cpp.

7.69.3.16 show()

```
std::string LSODE::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 20 of file LSODE.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/LSODE.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/LSODE.cpp

7.70 MainWindow Class Reference

```
#include <mainwindow.h>
```

Inheritance diagram for MainWindow:

Collaboration diagram for MainWindow:

Public Member Functions

- MainWindow (QWidget *parent=0)
- ~MainWindow ()

7.70.1 Detailed Description

Definition at line 13 of file mainwindow.h.

7.70.2 Constructor & Destructor Documentation

7.70.2.1 MainWindow() `MainWindow::MainWindow ( QWidget * parent = 0 ) [explicit]`

Definition at line 23 of file mainwindow.cpp.

Here is the call graph for this function:

7.70.2.2 ~MainWindow() `MainWindow::~MainWindow ( )`

Definition at line 57 of file mainwindow.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.cpp

7.71 Ui::MainWindow Class Reference

```
#include <ui_mainwindow.h>
```

Inheritance diagram for Ui::MainWindow:

Collaboration diagram for Ui::MainWindow:

Additional Inherited Members

7.71.1 Detailed Description

Definition at line 742 of file ui_mainwindow.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mainWindow/ui_mainwindow.h

7.72 MapKernelAndUI Class Reference

```
#include <mapkernelandui.h>
```

Collaboration diagram for MapKernelAndUI:

Public Member Functions

- MapKernelAndUI ()
- void map (Model ∗model, QObject ∗uiObj)
- void map (ModelElement ∗elem, QObject ∗uiObj)
- void unmap (Model ∗model, QObject ∗uiObj)
- void unmap (ModelElement ∗elem, QObject ∗uiObj)
- QObject ∗ getUI (Model ∗model)
- QObject ∗ getUI (ModelElement ∗elem)
- Model ∗ getModel (QObject ∗uiObj)
- ModelElement ∗ getElem (QObject ∗uiObj)

7.72.1 Detailed Description

Definition at line 9 of file mapkernelandui.h.

7.72.2 Constructor & Destructor Documentation

7.72.2.1 MapKernelAndUI() `MapKernelAndUI::MapKernelAndUI ( )`

Definition at line 4 of file mapkernelandui.cpp.

7.72.3 Member Function Documentation

7.72.3.1 getElem() `ModelElement * MapKernelAndUI::getElem ( QObject * uiObj )`

Definition at line 57 of file mapkernelandui.cpp.

7.72.3.2 getModel() `Model * MapKernelAndUI::getModel ( QObject * uiObj )`

Definition at line 50 of file mapkernelandui.cpp.

7.72.3.3 getUI() [1/2] `QObject * MapKernelAndUI::getUI ( Model * model )`

Definition at line 36 of file mapkernelandui.cpp.

7.72.3.4 getUI() [2/2] `QObject * MapKernelAndUI::getUI ( ModelElement * elem )`

Definition at line 43 of file mapkernelandui.cpp.

7.72.3.5 map() [1/2]

```
void MapKernelAndUI::map (
    Model * model,
    QObject * uiObj )
```

Definition at line 8 of file mapkernelandui.cpp.

7.72.3.6 map() [2/2]

```
void MapKernelAndUI::map (
    ModelElement * elem,
    QObject * uiObj )
```

Definition at line 13 of file mapkernelandui.cpp.

7.72.3.7 unmap() [1/2]

```
void MapKernelAndUI::unmap (
    Model * model,
    QObject * uiObj )
```

Definition at line 18 of file mapkernelandui.cpp.

7.72.3.8 unmap() [2/2]

```
void MapKernelAndUI::unmap (
    ModelElement * elem,
    QObject * uiObj )
```

Definition at line 27 of file mapkernelandui.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mapkernelandui.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mapkernelandui.cpp

7.73 MarkovChain Class Reference

```
#include <MarkovChain.h>
```

Inheritance diagram for MarkovChain:

Collaboration diagram for MarkovChain:

Public Member Functions

- MarkovChain (Model ∗model, std::string name="")
- virtual ∼MarkovChain ()=default
- virtual std::string show ()
- void setTransitionProbabilityMatrix (Variable ∗_transitionMatrix)
- Variable ∗ getTransitionMatrix () const
- Variable ∗ getCurrentState () const
- void setInitialDistribution (Variable ∗_initialDistribution)
- Variable ∗ getInitialState () const
- void setInititized (bool _inititized)
- bool isInititized () const
- void setCurrentState (Variable ∗_currentState)

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.73.1 Detailed Description

Definition at line 21 of file MarkovChain.h.

7.73.2 Constructor & Destructor Documentation

7.73.2.1 MarkovChain()

```
MarkovChain::MarkovChain (
    Model * model,
    std::string name = "" )
```

Definition at line 22 of file MarkovChain.cpp.

Here is the caller graph for this function:

7.73.2.2 ~MarkovChain()

```
virtual MarkovChain::~MarkovChain ( ) [virtual], [default]
```

7.73.3 Member Function Documentation

7.73.3.1 _check()

```
bool MarkovChain::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 127 of file MarkovChain.cpp.

7.73.3.2 _execute()

```
void MarkovChain::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 72 of file MarkovChain.cpp.

Here is the call graph for this function:

7.73.3.3 _initBetweenReplications()

```
void MarkovChain::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 117 of file MarkovChain.cpp.

7.73.3.4 _loadInstance()

```
bool MarkovChain::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 109 of file MarkovChain.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.73.3.5 _saveInstance() `std::map< std::string, std::string > * MarkovChain::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 121 of file MarkovChain.cpp.

Here is the call graph for this function:

7.73.3.6 getCurrentState() `Variable * MarkovChain::getCurrentState ( ) const`

Definition at line 48 of file MarkovChain.cpp.

7.73.3.7 getInitialState() `Variable * MarkovChain::getInitialState ( ) const`

Definition at line 60 of file MarkovChain.cpp.

7.73.3.8 GetPluginInformation() `PluginInformation * MarkovChain::GetPluginInformation ( ) [static]`

Definition at line 134 of file MarkovChain.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.73.3.9 getTransitionMatrix() `Variable * MarkovChain::getTransitionMatrix ( ) const`

Definition at line 44 of file MarkovChain.cpp.

7.73.3.10 isInitilized() `bool MarkovChain::isInitilized ( ) const`

Definition at line 68 of file MarkovChain.cpp.

7.73.3.11 LoadInstance() `ModelComponent * MarkovChain::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 30 of file MarkovChain.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.73.3.12 setCurrentState()

```
void MarkovChain::setCurrentState (
    Variable * _currentState )
```

Definition at line 52 of file MarkovChain.cpp.

7.73.3.13 setInitialDistribution()

```
void MarkovChain::setInitialDistribution (
    Variable * _initialDistribution )
```

Definition at line 56 of file MarkovChain.cpp.

7.73.3.14 setInitilized()

```
void MarkovChain::setInitilized (
    bool _initilized )
```

Definition at line 64 of file MarkovChain.cpp.

7.73.3.15 setTransitionProbabilityMatrix()

```
void MarkovChain::setTransitionProbabilityMatrix (
    Variable * _transitionMatrix )
```

Definition at line 40 of file MarkovChain.cpp.

7.73.3.16 show()

```
std::string MarkovChain::show ( )  [virtual]
```

Reimplemented from ModelComponent.

Definition at line 26 of file MarkovChain.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/MarkovChain.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/MarkovChain.cpp

7.74 Match Class Reference

```
#include <Match.h>
```

Inheritance diagram for Match:

Collaboration diagram for Match:

Public Member Functions

- Match (Model ∗model, std::string name="")
- virtual ∼Match ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.74.1 Detailed Description

Match module DESCRIPTION The Match module brings together a specified number of entities waiting in different queues. The match may be accomplished when there is at least one entity in each of the desired queues. Additionally, an attribute may be specified such that the entities waiting in the queues must have the same attribute values before the match is initiated. When an entity arrives at the Match module, it is placed in one of up to five associated queues, based on the entry point to which it is connected. Entities will remain in their respective queues until a match exists. Once a match exists, one entity from each queue is released to be matched. The matched entities are then synchronized to depart from the module. TYPICAL USES Assembling a part Gathering various products for a customer order Synchronizing a customer exit with a filled order Prompt Description Name Unique module identifier displayed on the module shape. Number to Match Number of matching entities that must reside in different queues before a match may be completed. Type Method for matching the incoming entities. If Type is Any Entities, one entity must reside in each queue for a match to be made. If Type is Based on Attribute, one entity must reside in each queue with the same attribute value. Attribute Name Attribute name that is used for identifying an arriving entity's match value. Applies only when Type is Based on Attribute.

Definition at line 47 of file Match.h.

7.74.2 Constructor & Destructor Documentation

7.74.2.1 Match()

```
Match::Match (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Match.cpp.

Here is the caller graph for this function:

7.74.2.2 ~Match()

```
virtual Match::~Match ( ) [virtual], [default]
```

7.74.3 Member Function Documentation

7.74.3.1 _check()

```
bool Match::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Match.cpp.

7.74.3.2 _execute()

```
void Match::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Match.cpp.

Here is the call graph for this function:

7.74.3.3 _initBetweenReplications()

```
void Match::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 48 of file Match.cpp.

7.74.3.4 _loadInstance()

```
bool Match::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 40 of file Match.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.74.3.5 _saveInstance()

```
std::map< std::string, std::string > * Match::_saveInstance ( ) [protected],
[virtual]
```

Reimplemented from ModelComponent.

Definition at line 51 of file Match.cpp.

Here is the call graph for this function:

7.74.3.6 GetPluginInformation() `PluginInformation * Match::GetPluginInformation ( ) [static]`

Definition at line 64 of file Match.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.74.3.7 LoadInstance() `ModelComponent * Match::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 25 of file Match.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.74.3.8 show() `std::string Match::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file Match.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Match.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Match.cpp

7.75 Model Class Reference

`#include <Model.h>`

Collaboration diagram for Model:

Public Member Functions

- Model (Simulator ∗simulator)
- virtual ∼Model ()=default
- bool save (std::string filename)
- bool load (std::string filename)
- bool check ()

  Checks the integrity and consistency of the model, possibly corrects some inconsistencies, and returns if the model is in position to the simulated.
- void clear ()
- void show ()
- bool insert (ModelElement ∗elemOrComp)

  Insert a new ModelElement or ModelComponent into the model (since 20191015). It's a generic access to ComponentManager->insert() or ModelElemento->insert()
- void remove (ModelElement ∗elemOrComp)

  Remove a new ModelElement or ModelComponent into the model (since 20191015). It's a generic access to ComponentManager->remove() or ModelElemento->remove()
- Entity ∗ createEntity (std::string name, bool insertIntoModel)
- void removeEntity (Entity ∗entity)
- void sendEntityToComponent (Entity ∗entity, Connection ∗connection, double timeDelay=0.0)

  Used by components (ModelComponent) to send entities to another specific component, usually the next one connected to it, or used by the model itself, when processing an event (Event).
- void sendEntityToComponent (Entity ∗entity, ModelComponent ∗component, double timeDelay=0.0, unsigned int componentInputNumber=0)

  Used by components (ModelComponent) to send entities to another specific component, usually the next one connected to it, or used by the model itself, when processing an event (Event).
- double parseExpression (const std::string expression)

  Invokes the parser to evaluate tyhe expression. Result is always a double, even if expression has syntatic errors (returns 0)
- double parseExpression (const std::string expression, bool ∗success, std::string ∗errorMessage)

  Invokes the parser to evaluate tyhe expression. Result is always a double, even if expression has syntatic errors (returns 0). Explicitly informs if there was an error.
- bool checkExpression (const std::string expression, const std::string expressionName, std::string ∗error↩ Message)

  This is invoked by ModelComponents and ModelElements in their private method _check() to verify if an expression defined by user is valid or not.
- Util::identification getId () const
- bool hasChanged () const
- OnEventManager ∗ getOnEvents () const

  Provides acccess to the class that manages events generated by the model, such as the beggining of a new simulation or replication, the processig of an event and much more.
- ElementManager ∗ getElements () const

  Provides access to the class that manages the most basic elements of the simulation model (such as queues, resources, variables, etc.).
- ComponentManager ∗ getComponents () const
- ModelInfo ∗ getInfos () const
- Simulator ∗ getParentSimulator () const
- ModelSimulation ∗ getSimulation () const

  Provides access to the class that manages the model simulation.
- List< Event ∗ > ∗ getFutureEvents () const

  The future events list chronologically sorted; Events are scheduled by components when processing other events, and a replication evolves over time by sequentially processing the very first event in this list. It's initialized with events first described by source components (SourceComponentModel).
- List< SimulationControl ∗ > ∗ getControls () const

Returns a list of values that can be externally controlled (changed). They usually correspond to input parameters in the simulation model that must be changed for an experimental design.

- List< SimulationResponse * > * getResponses () const

  Returns a list of exits or simulation results that can be read externally. They usually correspond to statistics resulting from the simulation that must be read for an experiment design.

- void setTracer (TraceManager *_traceManager)
- TraceManager * getTracer () const

  Provides access to the class that performs the trace of simulation and replications.

7.75.1 Detailed Description

Model is probably the most important class of Genesys kernel. It represents a discrete event-driven simulation model. Each model is responsible for controlling its own simulation, ie, for sequentially processing events and collecting statistical results. A model is mainly represented by a collection of components (ModelComponent), adequately configurated and connected, and a collection of under layered element (ModelElement).

Definition at line 44 of file Model.h.

7.75.2 Constructor & Destructor Documentation

7.75.2.1 Model()

```
Model::Model (
    Simulator * simulator )
```

The future events list must be chronologicaly sorted

Events are sorted chronologically

Definition at line 33 of file Model.cpp.

Here is the call graph for this function:

7.75.2.2 ~Model() `virtual Model::~Model ( ) [virtual], [default]`

Here is the caller graph for this function:

7.75.3 Member Function Documentation

7.75.3.1 check() `bool Model::check ( )`

Checks the integrity and consistency of the model, possibly corrects some inconsistencies, and returns if the model is in position to the simulated.

Definition at line 284 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.2 checkExpression()

```
bool Model::checkExpression (
    const std::string expression,
    const std::string expressionName,
    std::string * errorMessage )
```

This is invoked by ModelComponents and ModelElements in their private method _check() to verify if an expression defined by user is valid or not.

Definition at line 121 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.3 clear() `void Model::clear ( )`

Definition at line 231 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.4 createEntity() `Entity * Model::createEntity ( std::string name, bool insertIntoModel )`

Definition at line 305 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.5 getComponents() `ComponentManager * Model::getComponents ( ) const`

Definition at line 343 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.6 getControls() `List< SimulationControl * > * Model::getControls ( ) const`

Returns a list of values that can be externally controlled (changed). They usually correspond to input parameters in the simulation model that must be changed for an experimental design.

Definition at line 347 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.7 getElements() `ElementManager * Model::getElements ( ) const`

Provides access to the class that manages the most basic elements of the simulation model (such as queues, resources, variables, etc.).

Definition at line 359 of file Model.cpp.

7.75.3.8 getFutureEvents() `List< Event * > * Model::getFutureEvents ( ) const`

The future events list chronologically sorted; Events are scheduled by components when processing other events, and a replication evolves over time by sequentially processing the very first event in this list. It's initialized with events first described by source components (SourceComponentModel).

Definition at line 322 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.9 getId() `Util::identification Model::getId ( ) const`

Definition at line 375 of file Model.cpp.

7.75.3.10 getInfos() `ModelInfo * Model::getInfos ( ) const`

Definition at line 363 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.11 getOnEvents() `OnEventManager * Model::getOnEvents ( ) const`

Provides acccess to the class that manages events generated by the model, such as the beggining of a new simulation or replication, the processig of an event and much more.

Definition at line 355 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.12 getParentSimulator() `Simulator * Model::getParentSimulator ( ) const`

Definition at line 367 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.13 getResponses() `List< SimulationResponse * > * Model::getResponses ( ) const`

Returns a list of exits or simulation results that can be read externally. They usually correspond to statistics resulting from the simulation that must be read for an experiment design.

Definition at line 351 of file Model.cpp.

Here is the caller graph for this function:

7.75.3.14 getSimulation() `ModelSimulation * Model::getSimulation ( ) const`

Provides access to the class that manages the model simulation.

Definition at line 371 of file Model.cpp.

7.75.3.15 getTracer() `TraceManager * Model::getTracer ( ) const`

Provides access to the class that performs the trace of simulation and replications.

Definition at line 330 of file Model.cpp.

7.75.3.16 hasChanged() `bool Model::hasChanged ( ) const`

Definition at line 334 of file Model.cpp.

Here is the call graph for this function:

7.75.3.17 insert() `bool Model::insert ( ModelElement * elemOrComp )`

Insert a new ModelElement or ModelComponent into the model (since 20191015). It's a generic access to ComponentManager->insert() or ModelElemento->insert()

Definition at line 160 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.18 load()

```
bool Model::load (
    std::string filename )
```

Definition at line 103 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.19 parseExpression() [1/2]

```
double Model::parseExpression (
    const std::string expression )
```

Invokes the parser to evaluate tyhe expression. Result is always a double, even if expression has syntatic errors (returns 0)

Definition at line 113 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.20 parseExpression() [2/2]

```
double Model::parseExpression (
    const std::string expression,
    bool * success,
    std::string * errorMessage )
```

Invokes the parser to evaluate tyhe expression. Result is always a double, even if expression has syntatic errors (returns 0). Explicitly informs if there was an error.

Definition at line 137 of file Model.cpp.

Here is the call graph for this function:

7.75.3.21 remove()

```
void Model::remove (
    ModelElement * elemOrComp )
```

Remove a new ModelElement or ModelComponent into the model (since 20191015). It's a generic access to ComponentManager->remove() or ModelElemento->remove()

Definition at line 168 of file Model.cpp.

Here is the call graph for this function:

7.75.3.22 removeEntity()

```
void Model::removeEntity (
    Entity * entity )
```

Definition at line 314 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.23 save()

```
bool Model::save (
    std::string filename )
```

Definition at line 92 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.75.3.24 sendEntityToComponent() [1/2]

```
void Model::sendEntityToComponent (
    Entity * entity,
    Connection * connection,
    double timeDelay = 0.0 )
```

Used by components (ModelComponent) to send entities to another specific component, usually the next one connected to it, or used by the model itself, when processing an event (Event).

Definition at line 71 of file Model.cpp.

7.75.3.25 sendEntityToComponent() [2/2]

```
void Model::sendEntityToComponent (
    Entity * entity,
    ModelComponent * component,
    double timeDelay = 0.0,
    unsigned int componentInputNumber = 0 )
```

Used by components (ModelComponent) to send entities to another specific component, usually the next one connected to it, or used by the model itself, when processing an event (Event).

Definition at line 75 of file Model.cpp.

Here is the call graph for this function:

7.75.3.26 setTracer()

```
void Model::setTracer (
    TraceManager * _traceManager )
```

Definition at line 326 of file Model.cpp.

7.75.3.27 show() `void Model::show ( )`

Definition at line 142 of file Model.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Model.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Model.cpp

7.76 ModelChecker_if Class Reference

```
#include <ModelChecker_if.h>
```

Inheritance diagram for ModelChecker_if:

Collaboration diagram for ModelChecker_if:

Public Member Functions

- virtual bool checkAll ()=0

- virtual bool checkConnected ()=0
- virtual bool checkSymbols ()=0
- virtual bool checkActivationCode ()=0
- virtual bool checkLimits ()=0

7.76.1 Detailed Description

The ModelChecker is responsable for verifying the model consistency, fixing inconsistencies wheneaver possible

Definition at line 26 of file ModelChecker_if.h.

7.76.2 Member Function Documentation

7.76.2.1 checkActivationCode() `virtual bool ModelChecker_if::checkActivationCode ( ) [pure virtual]`

Checks if user-defined strings for symbols required by components, usually expressions or functions, are valid or references existing and valid elements.

Implemented in ModelCheckerDefaultImpl1.

7.76.2.2 checkAll() `virtual bool ModelChecker_if::checkAll ( ) [pure virtual]`

Implemented in ModelCheckerDefaultImpl1.

Here is the caller graph for this function:

7.76.2.3 checkConnected() `virtual bool ModelChecker_if::checkConnected ( ) [pure virtual]`

Invoques all other checks and returns true only if all of them returned true

Implemented in ModelCheckerDefaultImpl1.

7.76.2.4 checkLimits() `virtual bool ModelChecker_if::checkLimits ( ) [pure virtual]`

Checks if the installed version has acquired a valid activation code for commercial use

Implemented in ModelCheckerDefaultImpl1.

7.76.2.5 checkSymbols() `virtual bool ModelChecker_if::checkSymbols ( ) [pure virtual]`

Checks if components are consistently connected to other to form a valid process-oriented model, describing how entities proceed to the flow

Implemented in ModelCheckerDefaultImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelChecker_if.h

7.77 ModelCheckerDefaultImpl1 Class Reference

```
#include <ModelCheckerDefaultImpl1.h>
```

Inheritance diagram for ModelCheckerDefaultImpl1:

Collaboration diagram for ModelCheckerDefaultImpl1:

Public Member Functions

- ModelCheckerDefaultImpl1 (Model *model)
- virtual ~ModelCheckerDefaultImpl1 ()=default
- virtual bool checkAll ()
- virtual bool checkConnected ()
- virtual bool checkSymbols ()
- virtual bool checkActivationCode ()
- virtual bool checkLimits ()

7.77.1 Detailed Description

Definition at line 23 of file ModelCheckerDefaultImpl1.h.

7.77.2 Constructor & Destructor Documentation

7.77.2.1 ModelCheckerDefaultImpl1() `ModelCheckerDefaultImpl1::ModelCheckerDefaultImpl1 ( Model * model )`

Definition at line 24 of file ModelCheckerDefaultImpl1.cpp.

7.77.2.2 ~ModelCheckerDefaultImpl1() `virtual ModelCheckerDefaultImpl1::~ModelCheckerDefaultImpl1 ( ) [virtual], [default]`

7.77.3 Member Function Documentation

7.77.3.1 checkActivationCode() `bool ModelCheckerDefaultImpl1::checkActivationCode ( ) [virtual]`

Checks if user-defined strings for symbols required by components, usually expressions or functions, are valid or references existing and valid elements.

Implements ModelChecker_if.

Definition at line 172 of file ModelCheckerDefaultImpl1.cpp.

Here is the call graph for this function:

7.77.3.2 checkAll() `bool ModelCheckerDefaultImpl1::checkAll ( ) [virtual]`

Implements ModelChecker_if.

Definition at line 28 of file ModelCheckerDefaultImpl1.cpp.

Here is the call graph for this function:

7.77.3.3 checkConnected() `bool ModelCheckerDefaultImpl1::checkConnected ( ) [virtual]`

Invoques all other checks and returns true only if all of them returned true

Implements ModelChecker_if.

Definition at line 80 of file ModelCheckerDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.77.3.4 checkLimits() `bool ModelCheckerDefaultImpl1::checkLimits ( ) [virtual]`

Checks if the installed version has acquired a valid activation code for commercial use

Implements ModelChecker_if.

Definition at line 183 of file ModelCheckerDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.77.3.5 checkSymbols() `bool ModelCheckerDefaultImpl1::checkSymbols ( ) [virtual]`

Checks if components are consistently connected to other to form a valid process-oriented model, describing how entities proceed to the flow

Implements ModelChecker_if.

Definition at line 115 of file ModelCheckerDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.cpp

7.78 ModelComponent Class Reference

```
#include <ModelComponent.h>
```

Inherits ModelElement.

Inherited by Access, Assign, Batch, CelularAutomata, Decide, Delay, DropOff, Dummy, Enter, Exit, Hold, LSODE, Leave, MarkovChain, Match, PickStation, PickUp, Process, Record, Release, Remove, Route, Search, Seize, Separate, Signal, SinkModelComponent, SourceModelComponent, Start, Stop, Store, Submodel, Unstore, and Write.

Collaboration diagram for ModelComponent:

Classes

- struct DEFAULT_VALUES

Public Member Functions

- ModelComponent (Model ∗model, std::string componentTypename, std::string name="")
- virtual ∼ModelComponent ()
- virtual std::string show ()
- ConnectionManager ∗ getConnections () const

  Returns a list of components directly connected to the output. Usually the components have a single output, but they may have none (such as Dispose) or more than one (as Decide). In addition to the component, NextComponents specifies the inputNumber of the next component where the entity will be sent to. Ussually the components have a single input, but they may have none (such as Create) or more than one (as Match).
- void setDescription (std::string _description)
- std::string getDescription () const

Static Public Member Functions

- static void Execute (Entity ∗entity, ModelComponent ∗component, unsigned int inputNumber)

  This method triggers the simulation of the behavior of the component. It is invoked when an event (corresponding to this component) is taken from the list of future events or when an entity arrives at this component by connection.
- static void CreateInternalElements (ModelComponent ∗component)
- static bool Check (ModelComponent ∗component)
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)
- static std::map< std::string, std::string > ∗ SaveInstance (ModelComponent ∗component)

Protected Member Functions

- virtual void _execute (Entity *entity)=0
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Protected Attributes

- const struct ModelComponent::DEFAULT_VALUES DEFAULT
- std::string _description = DEFAULT.description
- ConnectionManager * _connections = new ConnectionManager()

7.78.1 Detailed Description

A component of the model is a block that represents a specific behavior to be simulated. The behavior is triggered when an entity arrives at the component, which corresponds to the occurrence of an event. A simulation model corresponds to a set of interconnected components to form the process by which the entity is submitted.

Parameters

model	The model this component belongs to

Definition at line 33 of file ModelComponent.h.

7.78.2 Constructor & Destructor Documentation

7.78.2.1 ModelComponent()

```
ModelComponent::ModelComponent (
    Model * model,
    std::string componentTypename,
    std::string name = "" )
```

Definition at line 19 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.2.2 ~ModelComponent() `ModelComponent::~ModelComponent ( ) [virtual]`

Definition at line 23 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.3 Member Function Documentation

7.78.3.1 _createInternalElements() `void ModelComponent::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelElement.

Reimplemented in Route, Record, Process, Leave, Enter, Dispose, Delay, Decide, and Create.

Definition at line 139 of file ModelComponent.cpp.

Here is the caller graph for this function:

7.78.3.2 _execute()

```
virtual void ModelComponent::_execute (
    Entity * entity ) [protected], [pure virtual]
```

Implemented in Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, and Access.

Here is the caller graph for this function:

7.78.3.3 _loadInstance()

```
bool ModelComponent::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Reimplemented in Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, SourceModelComponent, and SinkModelComponent.

Definition at line 102 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.3.4 _saveInstance() `std::map< std::string, std::string > * ModelComponent::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Reimplemented in Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, SourceModelComponent, and SinkModelComponent.

Definition at line 118 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.3.5 Check() `bool ModelComponent::Check ( ModelComponent * component ) [static]`

Definition at line 75 of file ModelComponent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.78.3.6 CreateInternalElements()

```
void ModelComponent::CreateInternalElements (
    ModelComponent * component ) [static]
```

Definition at line 44 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.3.7 Execute()

```
void ModelComponent::Execute (
    Entity * entity,
    ModelComponent * component,
    unsigned int inputNumber ) [static]
```

This method triggers the simulation of the behavior of the component. It is invoked when an event (corresponding to this component) is taken from the list of future events or when an entity arrives at this component by connection.

Definition at line 27 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.3.8 getConnections() `ConnectionManager * ModelComponent::getConnections ( ) const`

Returns a list of components directly connected to the output. Usually the components have a single output, but they may have none (such as Dispose) or more than one (as Decide). In addition to the component, NextComponents specifies the inputNumber of the next component where the entity will be sent to. Ussually the components have a single input, but they may have none (such as Create) or more than one (as Match).

Definition at line 94 of file ModelComponent.cpp.

7.78.3.9 getDescription() `std::string ModelComponent::getDescription ( ) const`

Definition at line 71 of file ModelComponent.cpp.

Here is the caller graph for this function:

7.78.3.10 LoadInstance()

```
static ModelComponent* ModelComponent::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

7.78.3.11 SaveInstance()

```
std::map< std::string, std::string > * ModelComponent::SaveInstance (
    ModelComponent * component ) [static]
```

Definition at line 53 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.3.12 setDescription()

```
void ModelComponent::setDescription (
    std::string _description )
```

Definition at line 64 of file ModelComponent.cpp.

Here is the caller graph for this function:

7.78.3.13 show() `std::string ModelComponent::show ( ) [virtual]`

Reimplemented from ModelElement.

Reimplemented in Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, and SourceModelComponent.

Definition at line 98 of file ModelComponent.cpp.

Here is the call graph for this function:

7.78.4 Member Data Documentation

7.78.4.1 _connections `ConnectionManager* ModelComponent::_connections = new ConnectionManager() [protected]`

Definition at line 64 of file ModelComponent.h.

7.78.4.2 _description `std::string ModelComponent::_description = DEFAULT.description [protected]`

Definition at line 63 of file ModelComponent.h.

7.78.4.3 DEFAULT `const struct ModelComponent::DEFAULT_VALUES ModelComponent::DEFAULT [protected]`

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelComponent.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelComponent.cpp

7.79 ModelElement Class Reference

```
#include <ModelElement.h>
```

Inheritance diagram for ModelElement:

Collaboration diagram for ModelElement:

Public Member Functions

- ModelElement (Model *model, std::string elementTypename, std::string name="", bool insertIntoModel=true)
- virtual ~ModelElement ()
- Util::identification getId () const
- void setName (std::string name)
- std::string getName () const
- std::string getClassname () const
- bool isReportStatistics () const
- void setReportStatistics (bool reportStatistics)
- virtual std::string show ()
- std::list< std::string > * getChildrenElementKeys () const
- ModelElement * getChildElement (std::string key) const
- bool hasChanged () const

Static Public Member Functions

- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields, bool insert↩ IntoModel)
- static std::map< std::string, std::string > * SaveInstance (ModelElement *element)
- static bool Check (ModelElement *element, std::string *errorMessage)
- static void CreateInternalElements (ModelElement *element)
- static void InitBetweenReplications (ModelElement *element)

Protected Member Functions

- void _setChildElement (std::string key, ModelElement *child)
- void _removeChildrenElements ()
- void _removeChildElement (std::string key)
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual ParserChangesInformation * _getParserChangesInformation ()
- virtual void _initBetweenReplications ()
- virtual void _createInternalElements ()

Protected Attributes

- Util::identification _id
- std::string _typename
- bool _reportStatistics
- bool _hasChanged
- Model * _parentModel
- std::map< std::string, ModelElement * > * _childrenElements = new std::map<std::string, ModelElement*>()

7.79.1 Detailed Description

This class is the basis for any element of the model (such as Queue, Resource, Variable, etc.) and also for any component of the model. It has the infrastructure to read and write on file and to verify symbols.

Definition at line 33 of file ModelElement.h.

7.79.2 Constructor & Destructor Documentation

7.79.2.1 ModelElement()

```
ModelElement::ModelElement (
    Model * model,
    std::string elementTypename,
    std::string name = "",
    bool insertIntoModel = true )
```

Definition at line 22 of file ModelElement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.79.2.2 ~ModelElement()

```
ModelElement::~ModelElement ( ) [virtual]
```

Definition at line 49 of file ModelElement.cpp.

Here is the call graph for this function:

7.79.3 Member Function Documentation

7.79.3.1 _check() `bool ModelElement::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented in Variable, Storage, Station, Set, Sequence, Schedule, Resource, Queue, File, Failure, EntityGroup, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, OLD_ODEelement, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, StatisticsCollector, SourceModelComponent, SinkModelComponent, EntityType, Entity, Counter, and Attribute.

Definition at line 107 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.2 _createInternalElements() `void ModelElement::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist before checking such expression

Reimplemented in Station, Resource, Queue, EntityGroup, Route, Record, Process, Leave, Enter, Dispose, Delay, Decide, Create, ModelComponent, and EntityType.

Definition at line 257 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.3 _getParserChangesInformation() `ParserChangesInformation * ModelElement::_getParser↩ ChangesInformation ( ) [protected], [virtual]`

This method returns all changes in the parser that are needed by plugins of this ModelElements. When connecting a new plugin, ParserChangesInformation are used to change parser source code, whch is after compiled and dinamically linked to to simulator kernel to reflect the changes

Reimplemented in Storage, Set, Queue, File, and Failure.

Definition at line 112 of file ModelElement.cpp.

7.79.3.4 _initBetweenReplications() `void ModelElement::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented in Variable, Resource, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, PickUp, PickStation, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, SourceModelComponent, SinkModelComponent, EntityType, and Counter.

Definition at line 116 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.5 _loadInstance() `bool ModelElement::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Implements PersistentObject_base.

Reimplemented in Variable, Storage, Station, Set, Sequence, Schedule, Resource, Queue, File, Failure, EntityGroup, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, OLD_ODEelement, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, StatisticsCollector, SourceModelComponent, SinkModelComponent, ModelComponent, EntityType, Entity, Counter, and Attribute.

Definition at line 85 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.6 _removeChildElement()

`void ModelElement::_removeChildElement ( std::string key ) [protected]`

Definition at line 68 of file ModelElement.cpp.

Here is the call graph for this function:

7.79.3.7 _removeChildrenElements() `void ModelElement::_removeChildrenElements ( ) [protected]`

Definition at line 55 of file ModelElement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.79.3.8 _saveInstance() `std::map< std::string, std::string > * ModelElement::_saveInstance ( ) [protected], [virtual]`

Implements PersistentObject_base.

Reimplemented in Variable, Storage, Station, Set, Sequence, Schedule, Resource, Queue, File, Failure, EntityGroup, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, OLD_ODEelement, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, StatisticsCollector, SourceModelComponent, SinkModelComponent, ModelComponent, EntityType, Entity, Counter, and Attribute.

Definition at line 98 of file ModelElement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.79.3.9 _setChildElement()

```
void ModelElement::_setChildElement (
    std::string key,
    ModelElement * child ) [protected]
```

7.79.3.10 Check()

```
bool ModelElement::Check (
    ModelElement * element,
    std::string * errorMessage ) [static]
```

This class method takes an instance of a ModelElement and invokes the private method_check() method of that instance, which checks itself

Definition at line 229 of file ModelElement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.79.3.11 CreateInternalElements()

```
void ModelElement::CreateInternalElements (
    ModelElement * element ) [static]
```

This class method is responsible for invoking the protected method _check() of the instance element, which creates any internal ModelElement (such as childrenElements) or even other external needed ModelElements, such as attributes or variables

Definition at line 247 of file ModelElement.cpp.

Here is the call graph for this function:

7.79.3.12 getChildElement()

```
ModelElement* ModelElement::getChildElement ( std::string key ) const
```

7.79.3.13 getChildrenElementKeys()

```
std::list< std::string > * ModelElement::getChildrenElementKeys ( ) const
```

Returns a list of keys (names) of internal ModelElements, cuch as Counters, StatisticsCollectors and others. ChildrenElements are ModelElements used by this ModelElement thar are needed before model checking

Definition at line 140 of file ModelElement.cpp.

7.79.3.14 getClassname()

```
std::string ModelElement::getClassname ( ) const
```

Definition at line 189 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.15 getId() `Util::identification ModelElement::getId ( ) const`

Definition at line 145 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.16 getName() `std::string ModelElement::getName ( ) const`

Definition at line 185 of file ModelElement.cpp.

7.79.3.17 hasChanged() `bool ModelElement::hasChanged ( ) const`

Definition at line 39 of file ModelElement.cpp.

7.79.3.18 InitBetweenReplications() `void ModelElement::InitBetweenReplications ( ModelElement * element ) [static]`

Definition at line 193 of file ModelElement.cpp.

Here is the call graph for this function:

7.79.3.19 isReportStatistics() `bool ModelElement::isReportStatistics ( ) const`

Return true if this ModelElement generates statics for simulation reports

Definition at line 268 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.3.20 LoadInstance() `ModelElement * ModelElement::LoadInstance ( Model * model, std::map< std::string, std::string > * fields, bool insertIntoModel ) [static]`

This class method receives a map of fields readed from a file (or somewhere else) creates an instace of the Model↩ Elements and inokes the protected method _loadInstance() of that instance, whch fills the field values. The instance can be automatticaly inserted into the simulation model if required

Definition at line 202 of file ModelElement.cpp.

Here is the call graph for this function:

7.79.3.21 SaveInstance()

```
std::map< std::string, std::string > * ModelElement::SaveInstance (
    ModelElement * element ) [static]
```

This class method takes an instance of a ModelElement, invokes the protected method _saveInstance() of that instance and retorns a map of filds (name=value) that can be saved on a file (or somewhere else)

Definition at line 219 of file ModelElement.cpp.

Here is the call graph for this function:

7.79.3.22 setName()

```
void ModelElement::setName (
    std::string name )
```

Definition at line 149 of file ModelElement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.79.3.23 setReportStatistics() `void ModelElement::setReportStatistics ( bool reportStatistics )`

Defnes if this ModelElement generates statics for simulation reports

Definition at line 261 of file ModelElement.cpp.

7.79.3.24 show() `std::string ModelElement::show ( ) [virtual]`

Reimplemented in Variable, Storage, Station, Set, Sequence, Schedule, Resource, Queue, File, Failure, EntityGroup, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, OLD_ODEelement, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, StatisticsCollector, SourceModelComponent, ModelComponent, EntityType, Entity, Counter, and Attribute.

Definition at line 128 of file ModelElement.cpp.

Here is the caller graph for this function:

7.79.4 Member Data Documentation

7.79.4.1 _childrenElements

`std::map<std::string, ModelElement*>* ModelElement::_childrenElements = new std::map<std::string, ModelElement*>()` [protected]

Definition at line 91 of file ModelElement.h.

7.79.4.2 _hasChanged `bool ModelElement::_hasChanged [protected]`

Definition at line 88 of file ModelElement.h.

7.79.4.3 _id `Util::identification ModelElement::_id [protected]`

Definition at line 85 of file ModelElement.h.

7.79.4.4 _parentModel `Model* ModelElement::_parentModel [protected]`

Definition at line 89 of file ModelElement.h.

7.79.4.5 _reportStatistics `bool ModelElement::_reportStatistics [protected]`

Definition at line 87 of file ModelElement.h.

7.79.4.6 _typename `std::string ModelElement::_typename [protected]`

Definition at line 86 of file ModelElement.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelElement.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelElement.cpp

7.80 ModelInfo Class Reference

```
#include <ModelInfo.h>
```

Collaboration diagram for ModelInfo:

Public Member Functions

- ModelInfo ()
- virtual ~ModelInfo ()=default
- std::string show ()
- void setName (std::string _name)
- std::string getName () const
- void setAnalystName (std::string _analystName)
- std::string getAnalystName () const
- void setDescription (std::string _description)
- std::string getDescription () const
- void setProjectTitle (std::string _projectTitle)
- std::string getProjectTitle () const
- void setVersion (std::string _version)
- std::string getVersion () const
- void loadInstance (std::map< std::string, std::string > *fields)
- std::map< std::string, std::string > * saveInstance ()
- bool hasChanged () const

7.80.1 Detailed Description

ModelInfo stores basic model project information, such as the project and analyst names.

Definition at line 25 of file ModelInfo.h.

7.80.2 Constructor & Destructor Documentation

7.80.2.1 ModelInfo() `ModelInfo::ModelInfo ( )`

Definition at line 19 of file ModelInfo.cpp.

7.80.2.2 ~ModelInfo() `virtual ModelInfo::~ModelInfo ( ) [virtual], [default]`

7.80.3 Member Function Documentation

7.80.3.1 getAnalystName() `std::string ModelInfo::getAnalystName ( ) const`

Definition at line 44 of file ModelInfo.cpp.

7.80.3.2 getDescription() `std::string ModelInfo::getDescription ( ) const`

Definition at line 53 of file ModelInfo.cpp.

7.80.3.3 getName() `std::string ModelInfo::getName ( ) const`

Definition at line 35 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.4 getProjectTitle() `std::string ModelInfo::getProjectTitle ( ) const`

Definition at line 62 of file ModelInfo.cpp.

7.80.3.5 getVersion() `std::string ModelInfo::getVersion ( ) const`

Definition at line 71 of file ModelInfo.cpp.

7.80.3.6 hasChanged() `bool ModelInfo::hasChanged ( ) const`

Definition at line 98 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.7 loadInstance() `void ModelInfo::loadInstance ( std::map< std::string, std::string > * fields )`

Definition at line 75 of file ModelInfo.cpp.

Here is the call graph for this function:

7.80.3.8 saveInstance() `std::map< std::string, std::string > * ModelInfo::saveInstance ( )`

Definition at line 86 of file ModelInfo.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.80.3.9 setAnalystName() `void ModelInfo::setAnalystName ( std::string _analystName )`

Definition at line 39 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.10 setDescription()

```
void ModelInfo::setDescription (
    std::string _description )
```

Definition at line 48 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.11 setName()

```
void ModelInfo::setName (
    std::string _name )
```

Definition at line 30 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.12 setProjectTitle()

```
void ModelInfo::setProjectTitle (
    std::string _projectTitle )
```

Definition at line 57 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.13 setVersion() `void ModelInfo::setVersion ( std::string _version )`

Definition at line 66 of file ModelInfo.cpp.

Here is the caller graph for this function:

7.80.3.14 show() `std::string ModelInfo::show ( )`

Definition at line 23 of file ModelInfo.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelInfo.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelInfo.cpp

7.81 ModelManager Class Reference

`#include <ModelManager.h>`

Collaboration diagram for ModelManager:

Public Member Functions

- ModelManager (Simulator *simulator)
- virtual ~ModelManager ()=default
- Model * newModel ()
- void insert (Model *model)
- void remove (Model *model)
- void setCurrent (Model *model)
- bool saveModel (std::string filename)
- bool loadModel (std::string filename)
- bool createFromLanguage (std::string modelSpecification)
- unsigned int size ()
- Model * front ()
- Model * current ()
- Model * next ()

7.81.1 Detailed Description

Definition at line 22 of file ModelManager.h.

7.81.2 Constructor & Destructor Documentation

7.81.2.1 ModelManager()

```
ModelManager::ModelManager (
    Simulator * simulator )
```

Definition at line 25 of file ModelManager.cpp.

7.81.2.2 ~ModelManager()

```
virtual ModelManager::~ModelManager ( ) [virtual], [default]
```

7.81.3 Member Function Documentation

7.81.3.1 createFromLanguage()

```
bool ModelManager::createFromLanguage (
    std::string modelSpecification )
```

Definition at line 73 of file ModelManager.cpp.

Here is the call graph for this function:

7.81.3.2 current() `Model * ModelManager::current ( )`

Definition at line 97 of file ModelManager.cpp.

7.81.3.3 front() `Model * ModelManager::front ( )`

Definition at line 101 of file ModelManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.81.3.4 insert() `void ModelManager::insert ( Model * model )`

Definition at line 35 of file ModelManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.81.3.5 loadModel()

```
bool ModelManager::loadModel (
    std::string filename )
```

Definition at line 60 of file ModelManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.81.3.6 newModel()

```
Model * ModelManager::newModel ( )
```

Definition at line 30 of file ModelManager.cpp.

Here is the caller graph for this function:

7.81.3.7 next() `Model * ModelManager::next ( )`

Definition at line 105 of file ModelManager.cpp.

Here is the call graph for this function:

7.81.3.8 remove()

```
void ModelManager::remove (
    Model * model )
```

Definition at line 41 of file ModelManager.cpp.

Here is the call graph for this function:

7.81.3.9 saveModel()

```
bool ModelManager::saveModel (
    std::string filename )
```

Definition at line 54 of file ModelManager.cpp.

Here is the call graph for this function:

7.81.3.10 setCurrent() `void ModelManager::setCurrent ( Model * model )`

Definition at line 93 of file ModelManager.cpp.

7.81.3.11 size() `unsigned int ModelManager::size ( )`

Definition at line 50 of file ModelManager.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelManager.cpp

7.82 ModelPersistence_if Class Reference

```
#include <ModelPersistence_if.h>
```

Inheritance diagram for ModelPersistence_if:

Collaboration diagram for ModelPersistence_if:

Public Member Functions

- virtual bool save (std::string filename)=0
- virtual bool load (std::string filename)=0
- virtual bool hasChanged ()=0

7.82.1 Detailed Description

First and inadequate interface for model persistence. It should use the best pattern for the DAO approach

Definition at line 22 of file ModelPersistence_if.h.

7.82.2 Member Function Documentation

7.82.2.1 hasChanged() `virtual bool ModelPersistence_if::hasChanged ( ) [pure virtual]`

Implemented in ModelPersistenceDefaultImpl1.

Here is the caller graph for this function:

7.82.2.2 load() `virtual bool ModelPersistence_if::load ( std::string filename ) [pure virtual]`

Implemented in ModelPersistenceDefaultImpl1.

Here is the caller graph for this function:

7.82.2.3 save() `virtual bool ModelPersistence_if::save ( std::string filename ) [pure virtual]`

Implemented in ModelPersistenceDefaultImpl1.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelPersistence_if.h

7.83 ModelPersistenceDefaultImpl1 Class Reference

```
#include <ModelPersistenceDefaultImpl1.h>
```

Inheritance diagram for ModelPersistenceDefaultImpl1:

Collaboration diagram for ModelPersistenceDefaultImpl1:

Public Member Functions

- ModelPersistenceDefaultImpl1 (Model *model)
- virtual ~ModelPersistenceDefaultImpl1 ()=default
- virtual bool save (std::string filename)
- virtual bool load (std::string filename)
- virtual bool hasChanged ()

7.83.1 Detailed Description

Definition at line 22 of file ModelPersistenceDefaultImpl1.h.

7.83.2 Constructor & Destructor Documentation

7.83.2.1 ModelPersistenceDefaultImpl1()

```
ModelPersistenceDefaultImpl1::ModelPersistenceDefaultImpl1 (
            Model * model )
```

Definition at line 28 of file ModelPersistenceDefaultImpl1.cpp.

7.83.2.2 ~ModelPersistenceDefaultImpl1()

```
virtual ModelPersistenceDefaultImpl1::~ModelPersistenceDefaultImpl1 ( ) [virtual], [default]
```

7.83.3 Member Function Documentation

7.83.3.1 hasChanged()

```
bool ModelPersistenceDefaultImpl1::hasChanged ( ) [virtual]
```

Implements ModelPersistence_if.

Definition at line 400 of file ModelPersistenceDefaultImpl1.cpp.

7.83.3.2 load() `bool ModelPersistenceDefaultImpl1::load ( std::string filename ) [virtual]`

Implements ModelPersistence_if.

Definition at line 253 of file ModelPersistenceDefaultImpl1.cpp.

Here is the call graph for this function:

7.83.3.3 save() `bool ModelPersistenceDefaultImpl1::save ( std::string filename ) [virtual]`

Implements ModelPersistence_if.

Definition at line 41 of file ModelPersistenceDefaultImpl1.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelPersistenceDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelPersistenceDefaultImpl1.cpp

7.84 ModelSimulation Class Reference

```
#include <ModelSimulation.h>
```

Collaboration diagram for ModelSimulation:

Public Member Functions

- ModelSimulation (Model *model)
- virtual ~ModelSimulation ()=default
- std::string show ()
- void start ()

  Starts a sequential execution of a simulation, ie, a set of replications of this model.
- void pause ()
- void step ()

  Executes the processing of a single event, the next one in the future events list.
- void stop ()
- void setNumberOfReplications (unsigned int _numberOfReplications)
- unsigned int getNumberOfReplications () const
- void setReplicationLength (double _replicationLength)
- double getReplicationLength () const
- void setReplicationLengthTimeUnit (Util::TimeUnit _replicationLengthTimeUnit)
- Util::TimeUnit getReplicationLengthTimeUnit () const
- void setReplicationReportBaseTimeUnit (Util::TimeUnit _replicationReportBaseTimeUnit)
- Util::TimeUnit getReplicationBaseTimeUnit () const
- void setWarmUpPeriod (double _warmUpPeriod)
- double getWarmUpPeriod () const
- void setWarmUpPeriodTimeUnit (Util::TimeUnit _warmUpPeriodTimeUnit)
- Util::TimeUnit getWarmUpPeriodTimeUnit () const
- void setTerminatingCondition (std::string _terminatingCondition)
- std::string getTerminatingCondition () const
- void setPauseOnEvent (bool _pauseOnEvent)
- bool isPauseOnEvent () const
- void setStepByStep (bool _stepByStep)
- bool isStepByStep () const

- void setInitializeStatistics (bool _initializeStatistics)
- bool isInitializeStatistics () const
- void setInitializeSystem (bool _initializeSystem)
- bool isInitializeSystem () const
- void setPauseOnReplication (bool _pauseBetweenReplications)
- bool isPauseOnReplication () const
- void setReporter (SimulationReporter_if *_simulationReporter)
- SimulationReporter_if * getReporter () const
- double getSimulatedTime () const
- bool isRunning () const
- bool isPaused () const
- unsigned int getCurrentReplicationNumber () const
- ModelComponent * getCurrentComponent () const
- Entity * getCurrentEntity () const
- unsigned int getCurrentInputNumber () const
- void setShowReportsAfterReplication (bool showReportsAfterReplication)
- bool isShowReportsAfterReplication () const
- void setShowReportsAfterSimulation (bool showReportsAfterSimulation)
- bool isShowReportsAfterSimulation () const
- List< double > * getBreakpointsOnTime () const
- List< Entity * > * getBreakpointsOnEntity () const
- List< ModelComponent * > * getBreakpointsOnComponent () const
- void loadInstance (std::map< std::string, std::string > *fields)
- std::map< std::string, std::string > * saveInstance ()
- Event * getCurrentEvent () const
- void setShowSimulationResposesInReport (bool _showSimulationResposesInReport)
- bool isShowSimulationResposesInReport () const
- void setShowSimulationControlsInReport (bool _showSimulationControlsInReport)
- bool isShowSimulationControlsInReport () const

Friends

- class Model

7.84.1 Detailed Description

The ModelSimulation controls the simulation of a model, alowing to start, pause, resume e stop a simulation, composed by a set of replications.

Definition at line 33 of file ModelSimulation.h.

7.84.2 Constructor & Destructor Documentation

7.84.2.1 ModelSimulation() `ModelSimulation::ModelSimulation ( Model * model )`

Definition at line 29 of file ModelSimulation.cpp.

Here is the call graph for this function:

7.84.2.2 ~ModelSimulation() `virtual ModelSimulation::~ModelSimulation ( ) [virtual], [default]`

7.84.3 Member Function Documentation

7.84.3.1 getBreakpointsOnComponent() `List< ModelComponent * > * ModelSimulation::getBreakpoints↩ OnComponent ( ) const`

Definition at line 580 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.2 getBreakpointsOnEntity() `List< Entity * > * ModelSimulation::getBreakpointsOnEntity ( ) const`

Definition at line 576 of file ModelSimulation.cpp.

7.84.3.3 getBreakpointsOnTime() `List< double > * ModelSimulation::getBreakpointsOnTime ( ) const`

Definition at line 572 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.4 getCurrentComponent() `ModelComponent * ModelSimulation::getCurrentComponent ( ) const`

Definition at line 536 of file ModelSimulation.cpp.

7.84.3.5 getCurrentEntity() `Entity * ModelSimulation::getCurrentEntity ( ) const`

Definition at line 540 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.6 getCurrentEvent() `Event * ModelSimulation::getCurrentEvent ( ) const`

Definition at line 670 of file ModelSimulation.cpp.

7.84.3.7 getCurrentInputNumber() `unsigned int ModelSimulation::getCurrentInputNumber ( ) const`

Definition at line 552 of file ModelSimulation.cpp.

7.84.3.8 getCurrentReplicationNumber() `unsigned int ModelSimulation::getCurrentReplication↩ Number ( ) const`

Definition at line 532 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.9 getNumberOfReplications() `unsigned int ModelSimulation::getNumberOfReplications ( ) const`

Definition at line 593 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.10 getReplicationBaseTimeUnit() `Util::TimeUnit ModelSimulation::getReplicationBaseTime↩ Unit ( ) const`

Definition at line 694 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.11 getReplicationLength() `double ModelSimulation::getReplicationLength ( ) const`

Definition at line 602 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.12 getReplicationLengthTimeUnit() `Util::TimeUnit ModelSimulation::getReplicationLength↩ TimeUnit ( ) const`

Definition at line 611 of file ModelSimulation.cpp.

7.84.3.13 getReporter() `SimulationReporter_if * ModelSimulation::getReporter ( ) const`

Definition at line 548 of file ModelSimulation.cpp.

7.84.3.14 getSimulatedTime() `double ModelSimulation::getSimulatedTime ( ) const`

Definition at line 524 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.15 getTerminatingCondition() `std::string ModelSimulation::getTerminatingCondition ( ) const`

Definition at line 638 of file ModelSimulation.cpp.

7.84.3.16 getWarmUpPeriod() `double ModelSimulation::getWarmUpPeriod ( ) const`

Definition at line 620 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.17 getWarmUpPeriodTimeUnit() `Util::TimeUnit ModelSimulation::getWarmUpPeriodTimeUnit ( ) const`

Definition at line 629 of file ModelSimulation.cpp.

7.84.3.18 isInitializeStatistics() `bool ModelSimulation::isInitializeStatistics ( ) const`

Definition at line 496 of file ModelSimulation.cpp.

7.84.3.19 isInitializeSystem() `bool ModelSimulation::isInitializeSystem ( ) const`

Definition at line 504 of file ModelSimulation.cpp.

7.84.3.20 isPaused() `bool ModelSimulation::isPaused ( ) const`

Definition at line 584 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.21 isPauseOnEvent() `bool ModelSimulation::isPauseOnEvent ( ) const`

Definition at line 488 of file ModelSimulation.cpp.

7.84.3.22 isPauseOnReplication() `bool ModelSimulation::isPauseOnReplication ( ) const`

Definition at line 520 of file ModelSimulation.cpp.

7.84.3.23 isRunning() `bool ModelSimulation::isRunning ( ) const`

The current time in the model being simulated, i.e., the instant when the current event was triggered

Definition at line 528 of file ModelSimulation.cpp.

7.84.3.24 isShowReportsAfterReplication() `bool ModelSimulation::isShowReportsAfterReplication ( ) const`

Definition at line 560 of file ModelSimulation.cpp.

7.84.3.25 isShowReportsAfterSimulation() `bool ModelSimulation::isShowReportsAfterSimulation ( ) const`

Definition at line 568 of file ModelSimulation.cpp.

7.84.3.26 isShowSimulationControlsInReport() `bool ModelSimulation::isShowSimulationControlsIn↩Report ( ) const`

Definition at line 686 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.27 isShowSimulationResposesInReport() `bool ModelSimulation::isShowSimulationResposes↩InReport ( ) const`

Definition at line 678 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.28 isStepByStep() `bool ModelSimulation::isStepByStep ( ) const`

Definition at line 512 of file ModelSimulation.cpp.

7.84.3.29 loadInstance() `void ModelSimulation::loadInstance ( std::map< std::string, std::string > * fields )`

Definition at line 642 of file ModelSimulation.cpp.

Here is the call graph for this function:

7.84.3.30 pause() `void ModelSimulation::pause ( )`

Definition at line 466 of file ModelSimulation.cpp.

7.84.3.31 saveInstance() `std::map< std::string, std::string > * ModelSimulation::saveInstance ( )`

Definition at line 655 of file ModelSimulation.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.84.3.32 setInitializeStatistics() `void ModelSimulation::setInitializeStatistics ( bool _initializeStatistics )`

Definition at line 492 of file ModelSimulation.cpp.

7.84.3.33 setInitializeSystem() `void ModelSimulation::setInitializeSystem ( bool _initializeSystem )`

Definition at line 500 of file ModelSimulation.cpp.

7.84.3.34 setNumberOfReplications()

```
void ModelSimulation::setNumberOfReplications (
    unsigned int _numberOfReplications )
```

Definition at line 588 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.35 setPauseOnEvent()

```
void ModelSimulation::setPauseOnEvent (
    bool _pauseOnEvent )
```

Definition at line 484 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.36 setPauseOnReplication() `void ModelSimulation::setPauseOnReplication ( bool _pauseBetweenReplications )`

Definition at line 516 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.37 setReplicationLength() `void ModelSimulation::setReplicationLength ( double _replicationLength )`

Definition at line 597 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.38 setReplicationLengthTimeUnit()

```
void ModelSimulation::setReplicationLengthTimeUnit (
    Util::TimeUnit _replicationLengthTimeUnit )
```

Definition at line 606 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.39 setReplicationReportBaseTimeUnit()

```
void ModelSimulation::setReplicationReportBaseTimeUnit (
        Util::TimeUnit _replicationReportBaseTimeUnit )
```

Definition at line 690 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.40 setReporter()

```
void ModelSimulation::setReporter (
        SimulationReporter_if * _simulationReporter )
```

Definition at line 544 of file ModelSimulation.cpp.

7.84.3.41 setShowReportsAfterReplication()

```
void ModelSimulation::setShowReportsAfterReplication (
    bool showReportsAfterReplication )
```

Definition at line 556 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.42 setShowReportsAfterSimulation()

```
void ModelSimulation::setShowReportsAfterSimulation (
    bool showReportsAfterSimulation )
```

Definition at line 564 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.43 setShowSimulationControlsInReport()

```
void ModelSimulation::setShowSimulationControls↩
InReport (
    bool _showSimulationControlsInReport )
```

Definition at line 682 of file ModelSimulation.cpp.

7.84.3.44 setShowSimulationResposesInReport()

```
void ModelSimulation::setShowSimulationResposes↩
InReport (
    bool _showSimulationResposesInReport )
```

Definition at line 674 of file ModelSimulation.cpp.

7.84.3.45 setStepByStep() `void ModelSimulation::setStepByStep ( bool _stepByStep )`

Definition at line 508 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.46 setTerminatingCondition() `void ModelSimulation::setTerminatingCondition ( std::string _terminatingCondition )`

Definition at line 633 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.47 setWarmUpPeriod() `void ModelSimulation::setWarmUpPeriod ( double _warmUpPeriod )`

Definition at line 615 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.48 setWarmUpPeriodTimeUnit()

```
void ModelSimulation::setWarmUpPeriodTimeUnit (
    Util::TimeUnit _warmUpPeriodTimeUnit )
```

Definition at line 624 of file ModelSimulation.cpp.

Here is the caller graph for this function:

7.84.3.49 show()

```
std::string ModelSimulation::show ( )
```

Definition at line 38 of file ModelSimulation.cpp.

Here is the call graph for this function:

7.84.3.50 start()

```
void ModelSimulation::start ( )
```

Starts a sequential execution of a simulation, ie, a set of replications of this model.

Checks the model and if ok then initialize the simulation, execute repeatedly each replication and then show simulation statistics

Definition at line 90 of file ModelSimulation.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.84.3.51 step() `void ModelSimulation::step ( )`

Executes the processing of a single event, the next one in the future events list.

Definition at line 470 of file ModelSimulation.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.84.3.52 stop() `void ModelSimulation::stop ( )`

Definition at line 477 of file ModelSimulation.cpp.

7.84.4 Friends And Related Function Documentation

7.84.4.1 Model `friend class Model [friend]`

Definition at line 119 of file ModelSimulation.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelSimulation.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ModelSimulation.cpp

7.85 ParserManager::NewParser Struct Reference

```
#include <ParserManager.h>
```

Collaboration diagram for ParserManager::NewParser:

Public Attributes

- std::string bisonFilename
- std::string flexFilename
- std::string compiledParserFilename

7.85.1 Detailed Description

Definition at line 24 of file ParserManager.h.

7.85.2 Member Data Documentation

7.85.2.1 bisonFilename `std::string ParserManager::NewParser::bisonFilename`

Definition at line 25 of file ParserManager.h.

7.85.2.2 compiledParserFilename `std::string ParserManager::NewParser::compiledParserFilename`

Definition at line 27 of file ParserManager.h.

7.85.2.3 flexFilename `std::string ParserManager::NewParser::flexFilename`

Definition at line 26 of file ParserManager.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserManager.h

7.86 obj_t Class Reference

`#include <obj_t.h>`

Collaboration diagram for obj_t:

Public Member Functions

- obj_t ()
- virtual ~obj_t ()
- obj_t (double v, std::string t)
- obj_t (double v, std::string t, unsigned long uid)

Public Attributes

- double valor
- std::string tipo
- unsigned long id

7.86.1 Detailed Description

Definition at line 5 of file obj_t.h.

7.86.2 Constructor & Destructor Documentation

7.86.2.1 obj_t() [1/3] `obj_t::obj_t ( )`

Definition at line 4 of file obj_t.cpp.

7.86.2.2 ~obj_t() `obj_t::~obj_t ( ) [virtual]`

Definition at line 8 of file obj_t.cpp.

7.86.2.3 obj_t() [2/3]

```
obj_t::obj_t (
    double v,
    std::string t )
```

Definition at line 12 of file obj_t.cpp.

7.86.2.4 obj_t() [3/3]

```
obj_t::obj_t (
    double v,
    std::string t,
    unsigned long uid )
```

Definition at line 18 of file obj_t.cpp.

7.86.3 Member Data Documentation

7.86.3.1 id `unsigned long obj_t::id`

Definition at line 13 of file obj_t.h.

7.86.3.2 tipo `std::string obj_t::tipo`

Definition at line 12 of file obj_t.h.

7.86.3.3 valor `double obj_t::valor`

Definition at line 11 of file obj_t.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.cpp

7.87 ODEfunction Class Reference

`#include <OLD_ODEelement.h>`

Collaboration diagram for ODEfunction:

Public Member Functions

- ODEfunction (std::string expression, double initialPoint, double initialValue)

Public Attributes

- std::string expression
- double initialPoint
- double initialValue

7.87.1 Detailed Description

Definition at line 21 of file OLD_ODEelement.h.

7.87.2 Constructor & Destructor Documentation

7.87.2.1 ODEfunction()

```
ODEfunction::ODEfunction (
    std::string expression,
    double initialPoint,
    double initialValue ) [inline]
```

Definition at line 23 of file OLD_ODEelement.h.

7.87.3 Member Data Documentation

7.87.3.1 expression

`std::string ODEfunction::expression`

Definition at line 28 of file OLD_ODEelement.h.

7.87.3.2 initialPoint

`double ODEfunction::initialPoint`

Definition at line 29 of file OLD_ODEelement.h.

7.87.3.3 initialValue

`double ODEfunction::initialValue`

Definition at line 30 of file OLD_ODEelement.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/OLD_ODEelement.h

7.88 OLD_ODEelement Class Reference

```
#include <OLD_ODEelement.h>
```

Inheritance diagram for OLD_ODEelement:

Collaboration diagram for OLD_ODEelement:

Public Member Functions

- OLD_ODEelement (Model ∗model, std::string name="")
- virtual ∼OLD_ODEelement ()=default
- virtual std::string show ()
- double solve ()
- void setStepH (double _h)
- double getStepH () const
- void setEndTime (double _endTime)
- double getEndTime () const
- List< ODEfunction ∗ > ∗ getODEfunctions () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.88.1 Detailed Description

Definition at line 33 of file OLD_ODEelement.h.

7.88.2 Constructor & Destructor Documentation

7.88.2.1 OLD_ODEelement()

```
OLD_ODEelement::OLD_ODEelement (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file OLD_ODEelement.cpp.

Here is the caller graph for this function:

7.88.2.2 ~OLD_ODEelement()

```
virtual OLD_ODEelement::~OLD_ODEelement ( ) [virtual], [default]
```

7.88.3 Member Function Documentation

7.88.3.1 _check()

```
bool OLD_ODEelement::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 78 of file OLD_ODEelement.cpp.

Here is the call graph for this function:

7.88.3.2 _loadInstance() `bool OLD_ODEelement::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 70 of file OLD_ODEelement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.88.3.3 _saveInstance() `std::map< std::string, std::string > * OLD_ODEelement::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 74 of file OLD_ODEelement.cpp.

Here is the call graph for this function:

7.88.3.4 getEndTime() `double OLD_ODEelement::getEndTime ( ) const`

Definition at line 47 of file OLD_ODEelement.cpp.

7.88.3.5 getODEfunctions() `List< ODEfunction * > * OLD_ODEelement::getODEfunctions ( ) const`

Definition at line 51 of file OLD_ODEelement.cpp.

7.88.3.6 GetPluginInformation() `PluginInformation * OLD_ODEelement::GetPluginInformation ( ) [static]`

Definition at line 55 of file OLD_ODEelement.cpp.

Here is the call graph for this function:

7.88.3.7 getStepH() `double OLD_ODEelement::getStepH ( ) const`

Definition at line 39 of file OLD_ODEelement.cpp.

7.88.3.8 LoadInstance() `ModelElement * OLD_ODEelement::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 60 of file OLD_ODEelement.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.88.3.9 setEndTime() `void OLD_ODEelement::setEndTime ( double _endTime )`

Definition at line 43 of file OLD_ODEelement.cpp.

7.88.3.10 setStepH() `void OLD_ODEelement::setStepH ( double _h )`

Definition at line 35 of file OLD_ODEelement.cpp.

7.88.3.11 show() `std::string OLD_ODEelement::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 21 of file OLD_ODEelement.cpp.

Here is the call graph for this function:

7.88.3.12 solve() `double OLD_ODEelement::solve ( )`

Definition at line 30 of file OLD_ODEelement.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/OLD_ODEelement.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/OLD_ODEelement.cpp

7.89 OnEventManager Class Reference

```
#include <OnEventManager.h>
```

Collaboration diagram for OnEventManager:

Public Member Functions

- OnEventManager ()
- virtual ~OnEventManager ()=default
- void addOnReplicationStartHandler (simulationEventHandler EventHandler)
- void addOnReplicationStepHandler (simulationEventHandler EventHandler)
- void addOnReplicationEndHandler (simulationEventHandler EventHandler)
- void addOnProcessEventHandler (simulationEventHandler EventHandler)
- void addOnEntityCreateHandler (simulationEventHandler EventHandler)
- void addOnEntityMoveHandler (simulationEventHandler EventHandler)
- void addOnEntityRemoveHandler (simulationEventHandler EventHandler)
- void addOnSimulationStartHandler (simulationEventHandler EventHandler)
- void addOnSimulationPausedHandler (simulationEventHandler EventHandler)
- void addOnSimulationResumeHandler (simulationEventHandler EventHandler)
- void addOnSimulationEndHandler (simulationEventHandler EventHandler)
- void addOnBreakpointHandler (simulationEventHandler EventHandler)
- template<typename Class >
  void addOnReplicationStartHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnReplicationStepHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnReplicationEndHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnProcessEventHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnEntityCreateHandler (Class *object, void(Class::*function)(SimulationEvent *))

- template<typename Class >
  void addOnEntityMoveHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnEntityRemoveHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnSimulationStartHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnSimulationPausedHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnSimulationResumeHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnSimulationEndHandler (Class *object, void(Class::*function)(SimulationEvent *))
- template<typename Class >
  void addOnBreakpointHandler (Class *object, void(Class::*function)(SimulationEvent *))
- void NotifyReplicationStartHandlers (SimulationEvent *se)
- void NotifyReplicationStepHandlers (SimulationEvent *se)
- void NotifyReplicationEndHandlers (SimulationEvent *se)
- void NotifyProcessEventHandlers (SimulationEvent *se)
- void NotifyEntityCreateHandlers (SimulationEvent *se)
- void NotifyEntityMoveHandlers (SimulationEvent *se)
- void NotifyEntityRemoveHandlers (SimulationEvent *se)
- void NotifySimulationStartHandlers (SimulationEvent *se)
- void NotifySimulationPausedHandlers (SimulationEvent *se)
- void NotifySimulationResumeHandlers (SimulationEvent *se)
- void NotifySimulationEndHandlers (SimulationEvent *se)
- void NotifyBreakpointHandlers (SimulationEvent *se)

7.89.1 Detailed Description

OnEventManager allows external methods to hook interval simulation events as listeners (or observers) of pecific events. All methods added as listeners of an event will be invovked when that event is triggered.

Definition at line 140 of file OnEventManager.h.

7.89.2 Constructor & Destructor Documentation

7.89.2.1 OnEventManager() `OnEventManager::OnEventManager ( )`

Definition at line 18 of file OnEventManager.cpp.

7.89.2.2 ~OnEventManager() `virtual OnEventManager::~OnEventManager ( ) [virtual], [default]`

7.89.3 Member Function Documentation

7.89.3.1 addOnBreakpointHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnBreakpointHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 269 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.2 addOnBreakpointHandler() [2/2] void OnEventManager::addOnBreakpointHandler (

```
            simulationEventHandler EventHandler )
```

Definition at line 70 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.3 addOnEntityCreateHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnEntityCreateHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 241 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.4 addOnEntityCreateHandler() [2/2]

```
void OnEventManager::addOnEntityCreateHandler (
        simulationEventHandler EventHandler )
```

Definition at line 38 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.5 addOnEntityMoveHandler() [1/2]

```
template<typename Class >
void OnEventManager::addOnEntityMoveHandler (
        Class * object,
        void(Class::*)(SimulationEvent *) function )
```

Definition at line 245 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.6 addOnEntityMoveHandler() [2/2]

```
void OnEventManager::addOnEntityMoveHandler (
        simulationEventHandler EventHandler )
```

Definition at line 42 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.7 addOnEntityRemoveHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnEntityRemoveHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 249 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.8 addOnEntityRemoveHandler() [2/2] void OnEventManager::addOnEntityRemoveHandler (

```
            simulationEventHandler EventHandler )
```

Definition at line 46 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.9 addOnProcessEventHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnProcessEventHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 237 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.10 addOnProcessEventHandler() [2/2]

```
void OnEventManager::addOnProcessEventHandler (
    simulationEventHandler EventHandler )
```

Definition at line 34 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.11 addOnReplicationEndHandler() [1/2]

```
template<typename Class >
void OnEventManager::addOnReplicationEndHandler (
    Class * object,
    void(Class::*)(SimulationEvent *) function )
```

Definition at line 233 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.12 addOnReplicationEndHandler() [2/2]

```
void OnEventManager::addOnReplicationEndHandler (
    simulationEventHandler EventHandler )
```

Definition at line 50 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.13 addOnReplicationStartHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnReplicationStartHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 225 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.14 addOnReplicationStartHandler() [2/2] void OnEventManager::addOnReplicationStartHandler (

```
            simulationEventHandler EventHandler )
```

Definition at line 26 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.15 addOnReplicationStepHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnReplicationStepHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 229 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.16 addOnReplicationStepHandler() [2/2]

```
void OnEventManager::addOnReplicationStepHandler (
            simulationEventHandler EventHandler )
```

Definition at line 30 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.17 addOnSimulationEndHandler() [1/2]

```
template<typename Class >
void OnEventManager::addOnSimulationEndHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 265 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.18 addOnSimulationEndHandler() [2/2]

```
void OnEventManager::addOnSimulationEndHandler (
            simulationEventHandler EventHandler )
```

Definition at line 66 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.19 addOnSimulationPausedHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnSimulationPausedHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 257 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.20 addOnSimulationPausedHandler() [2/2] void OnEventManager::addOnSimulationPaused↩ Handler (

```
            simulationEventHandler EventHandler )
```

Definition at line 58 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.21 addOnSimulationResumeHandler() [1/2] template<typename Class >

```
void OnEventManager::addOnSimulationResumeHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 261 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.22 addOnSimulationResumeHandler() [2/2]

```
void OnEventManager::addOnSimulationResume↩
Handler (
            simulationEventHandler EventHandler )
```

Definition at line 62 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.23 addOnSimulationStartHandler() [1/2]

```
template<typename Class >
void OnEventManager::addOnSimulationStartHandler (
            Class * object,
            void(Class::*)(SimulationEvent *) function )
```

Definition at line 253 of file OnEventManager.h.

Here is the call graph for this function:

7.89.3.24 addOnSimulationStartHandler() [2/2]

```
void OnEventManager::addOnSimulationStartHandler
(
            simulationEventHandler EventHandler )
```

Definition at line 54 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.25 NotifyBreakpointHandlers()

```
void OnEventManager::NotifyBreakpointHandlers (
    SimulationEvent * se )
```

Definition at line 141 of file OnEventManager.cpp.

7.89.3.26 NotifyEntityCreateHandlers()

```
void OnEventManager::NotifyEntityCreateHandlers (
    SimulationEvent * se )
```

Definition at line 101 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.27 NotifyEntityMoveHandlers()

```
void OnEventManager::NotifyEntityMoveHandlers (
    SimulationEvent * se )
```

Definition at line 106 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.28 NotifyEntityRemoveHandlers()

```
void OnEventManager::NotifyEntityRemoveHandlers (
    SimulationEvent * se )
```

Definition at line 111 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.29 NotifyProcessEventHandlers()

```
void OnEventManager::NotifyProcessEventHandlers (
    SimulationEvent * se )
```

Definition at line 116 of file OnEventManager.cpp.

7.89.3.30 NotifyReplicationEndHandlers()

```
void OnEventManager::NotifyReplicationEndHandlers (
    SimulationEvent * se )
```

Definition at line 96 of file OnEventManager.cpp.

7.89.3.31 NotifyReplicationStartHandlers()

```
void OnEventManager::NotifyReplicationStartHandlers (
    SimulationEvent * se )
```

Definition at line 86 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.32 NotifyReplicationStepHandlers()

```
void OnEventManager::NotifyReplicationStepHandlers (
    SimulationEvent * se )
```

Definition at line 91 of file OnEventManager.cpp.

7.89.3.33 NotifySimulationEndHandlers()

```
void OnEventManager::NotifySimulationEndHandlers (
    SimulationEvent * se )
```

Definition at line 136 of file OnEventManager.cpp.

7.89.3.34 NotifySimulationPausedHandlers()

```
void OnEventManager::NotifySimulationPausedHandlers
(
    SimulationEvent * se )
```

Definition at line 126 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.35 NotifySimulationResumeHandlers()

```
void OnEventManager::NotifySimulationResumeHandlers
(
    SimulationEvent * se )
```

Definition at line 131 of file OnEventManager.cpp.

Here is the caller graph for this function:

7.89.3.36 NotifySimulationStartHandlers()

```
void OnEventManager::NotifySimulationStartHandlers ( SimulationEvent * se )
```

Definition at line 121 of file OnEventManager.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/OnEventManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/OnEventManager.cpp

7.90 OperatingSystem02 Class Reference

```
#include <OperatingSystem02.h>
```

Inheritance diagram for OperatingSystem02:

Collaboration diagram for OperatingSystem02:

Public Member Functions

- OperatingSystem02 ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.90.1 Detailed Description

Definition at line 19 of file OperatingSystem02.h.

7.90.2 Constructor & Destructor Documentation

7.90.2.1 OperatingSystem02() `OperatingSystem02::OperatingSystem02 ( )`

Definition at line 35 of file OperatingSystem02.cpp.

7.90.3 Member Function Documentation

7.90.3.1 main()

```
int OperatingSystem02::main (
    int argc,
    char ** argv )  [virtual]
```

Implements BaseConsoleGenesysApplication.

Definition at line 38 of file OperatingSystem02.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/OperatingSystem02.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/OperatingSystem02.cpp

7.91 OperatingSystem03 Class Reference

```
#include <OperatingSystem03.h>
```

Inheritance diagram for OperatingSystem03:

Collaboration diagram for OperatingSystem03:

Public Member Functions

- OperatingSystem03 ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.91.1 Detailed Description

Definition at line 19 of file OperatingSystem03.h.

7.91.2 Constructor & Destructor Documentation

7.91.2.1 OperatingSystem03() `OperatingSystem03::OperatingSystem03 ( )`

Definition at line 36 of file OperatingSystem03.cpp.

7.91.3 Member Function Documentation

7.91.3.1 main()

```
int OperatingSystem03::main (
    int argc,
    char ** argv )  [virtual]
```

Implements BaseConsoleGenesysApplication.

Definition at line 39 of file OperatingSystem03.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/OperatingSystem03.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/teaching/OperatingSystem03.cpp

7.92 Parser_if Class Reference

```
#include <Parser_if.h>
```

Inheritance diagram for Parser_if:

Collaboration diagram for Parser_if:

Public Member Functions

- virtual double parse (const std::string expression)=0

- virtual double parse (const std::string expression, bool ∗success, std::string ∗errorMessage)=0
- virtual std::string ∗ getErrorMessage ()=0
- virtual void setSampler (Sampler_if ∗sampler)=0
- virtual Sampler_if ∗ sampler () const =0

7.92.1 Detailed Description

Definition at line 20 of file Parser_if.h.

7.92.2 Member Function Documentation

7.92.2.1 getErrorMessage() `virtual std::string* Parser_if::getErrorMessage ( ) [pure virtual]`

Implemented in ParserDefaultImpl2, and ParserDefaultImpl1.

7.92.2.2 parse() `[1/2] virtual double Parser_if::parse ( const std::string expression ) [pure virtual]`

Implemented in ParserDefaultImpl2, and ParserDefaultImpl1.

Here is the caller graph for this function:

7.92.2.3 parse() [2/2]

```
virtual double Parser_if::parse (
    const std::string expression,
    bool * success,
    std::string * errorMessage ) [pure virtual]
```

Implemented in ParserDefaultImpl2, and ParserDefaultImpl1.

7.92.2.4 sampler()

```
virtual Sampler_if* Parser_if::sampler ( ) const [pure virtual]
```

Implemented in ParserDefaultImpl2.

7.92.2.5 setSampler()

```
virtual void Parser_if::setSampler (
    Sampler_if * sampler ) [pure virtual]
```

Implemented in ParserDefaultImpl2.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Parser_if.h

7.93 ParserChangesInformation Class Reference

```
#include <ParserChangesInformation.h>
```

Collaboration diagram for ParserChangesInformation:

Public Member Functions

- ParserChangesInformation ()
- virtual ~ParserChangesInformation ()=default

7.93.1 Detailed Description

Definition at line 20 of file ParserChangesInformation.h.

7.93.2 Constructor & Destructor Documentation

7.93.2.1 ParserChangesInformation() `ParserChangesInformation::ParserChangesInformation ( )`

Definition at line 16 of file ParserChangesInformation.cpp.

7.93.2.2 ~ParserChangesInformation() `virtual ParserChangesInformation::~ParserChangesInformation ( ) [virtual], [default]`

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserChangesInformation.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserChangesInformation.cpp

7.94 ParserDefaultImpl1 Class Reference

`#include <ParserDefaultImpl1.h>`

Inheritance diagram for ParserDefaultImpl1:

Collaboration diagram for ParserDefaultImpl1:

Public Member Functions

- ParserDefaultImpl1 (Model ∗model)
- virtual ∼ParserDefaultImpl1 ()=default
- double parse (const std::string expression)
- double parse (const std::string expression, bool ∗success, std::string ∗errorMessage)
- std::string ∗ getErrorMessage ()

7.94.1 Detailed Description

Definition at line 21 of file ParserDefaultImpl1.h.

7.94.2 Constructor & Destructor Documentation

7.94.2.1 ParserDefaultImpl1() `ParserDefaultImpl1::ParserDefaultImpl1 ( Model * model )`

Definition at line 18 of file ParserDefaultImpl1.cpp.

7.94.2.2 ~ParserDefaultImpl1() `virtual ParserDefaultImpl1::~ParserDefaultImpl1 ( ) [virtual], [default]`

7.94.3 Member Function Documentation

7.94.3.1 getErrorMessage() `std::string * ParserDefaultImpl1::getErrorMessage ( ) [virtual]`

Implements Parser_if.

Definition at line 27 of file ParserDefaultImpl1.cpp.

7.94.3.2 parse() [1/2] `double ParserDefaultImpl1::parse ( const std::string expression ) [virtual]`

Implements Parser_if.

Definition at line 22 of file ParserDefaultImpl1.cpp.

Here is the caller graph for this function:

7.94.3.3 parse() [2/2] `double ParserDefaultImpl1::parse ( const std::string expression, bool * success, std::string * errorMessage ) [virtual]`

Implements Parser_if.

Definition at line 32 of file ParserDefaultImpl1.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserDefaultImpl1.cpp

7.95 ParserDefaultImpl2 Class Reference

```
#include <ParserDefaultImpl2.h>
```

Inheritance diagram for ParserDefaultImpl2:

Collaboration diagram for ParserDefaultImpl2:

Public Member Functions

- ParserDefaultImpl2 (Model ∗model, Sampler_if ∗sampler, bool throws=false)
- virtual ∼ParserDefaultImpl2 ()=default
- virtual double parse (const std::string expression)
- virtual double parse (const std::string expression, bool ∗success, std::string ∗errorMessage)
- virtual std::string ∗ getErrorMessage ()
- virtual void setSampler (Sampler_if ∗_sampler)
- virtual Sampler_if ∗ sampler () const

7.95.1 Detailed Description

Definition at line 25 of file ParserDefaultImpl2.h.

7.95.2 Constructor & Destructor Documentation

7.95.2.1 ParserDefaultImpl2()

```
ParserDefaultImpl2::ParserDefaultImpl2 (
    Model * model,
    Sampler_if * sampler,
    bool throws = false )
```

Definition at line 18 of file ParserDefaultImpl2.cpp.

Here is the call graph for this function:

7.95.2.2 ~ParserDefaultImpl2()

```
virtual ParserDefaultImpl2::~ParserDefaultImpl2 ( ) [virtual], [default]
```

7.95.3 Member Function Documentation

7.95.3.1 getErrorMessage()

```
std::string * ParserDefaultImpl2::getErrorMessage ( ) [virtual]
```

Implements Parser_if.

Definition at line 38 of file ParserDefaultImpl2.cpp.

Here is the call graph for this function:

7.95.3.2 parse() [1/2] double ParserDefaultImpl2::parse (const std::string *expression*) [virtual]

Implements Parser_if.

Definition at line 23 of file ParserDefaultImpl2.cpp.

Here is the call graph for this function:

7.95.3.3 parse() [2/2] double ParserDefaultImpl2::parse (const std::string *expression*, bool * *success*, std::string * *errorMessage*) [virtual]

Implements Parser_if.

Definition at line 43 of file ParserDefaultImpl2.cpp.

Here is the call graph for this function:

7.95.3.4 sampler() `Sampler_if * ParserDefaultImpl2::sampler ( ) const [virtual]`

Implements Parser_if.

Definition at line 65 of file ParserDefaultImpl2.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.95.3.5 setSampler() `void ParserDefaultImpl2::setSampler ( Sampler_if * _sampler ) [virtual]`

Implements Parser_if.

Definition at line 61 of file ParserDefaultImpl2.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserDefaultImpl2.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ParserDefaultImpl2.cpp

7.96 ParserManager Class Reference

```
#include <ParserManager.h>
```

Collaboration diagram for ParserManager:

Classes

- struct GenerateNewParserResult
- struct NewParser

Public Member Functions

- ParserManager ()
- virtual ~ParserManager ()=default
- ParserManager::GenerateNewParserResult generateNewParser (ParserChangesInformation *changes)
- bool connectNewParser (ParserManager::NewParser newParser)

7.96.1 Detailed Description

Definition at line 21 of file ParserManager.h.

7.96.2 Constructor & Destructor Documentation

7.96.2.1 ParserManager() `ParserManager::ParserManager ( )`

Definition at line 18 of file ParserManager.cpp.

7.96.2.2 ~ParserManager() `virtual ParserManager::~ParserManager ( ) [virtual], [default]`

7.96.3 Member Function Documentation

7.96.3.1 connectNewParser() `bool ParserManager::connectNewParser ( ParserManager::NewParser newParser )`

7.96.3.2 generateNewParser() `ParserManager::GenerateNewParserResult ParserManager::generateNewParser ( ParserChangesInformation * changes )`

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.cpp

7.97 PersistentObject_base Class Reference

`#include <PersistentObject_base.h>`

Inheritance diagram for PersistentObject_base:

Collaboration diagram for PersistentObject_base:

Public Member Functions

- PersistentObject_base ()
- virtual ~PersistentObject_base ()=default

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)=0
- virtual std::map< std::string, std::string > * _saveInstance ()=0

7.97.1 Detailed Description

Definition at line 20 of file PersistentObject_base.h.

7.97.2 Constructor & Destructor Documentation

7.97.2.1 PersistentObject_base() `PersistentObject_base::PersistentObject_base ( ) [inline]`

Definition at line 22 of file PersistentObject_base.h.

7.97.2.2 ~PersistentObject_base() `virtual PersistentObject_base::~PersistentObject_base ( ) [virtual], [default]`

7.97.3 Member Function Documentation

7.97.3.1 _loadInstance() `virtual bool PersistentObject_base::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [pure virtual]`

Implemented in Variable, Storage, Station, Set, Sequence, SequenceStep, Schedule, Resource, Queue, File, Failure, EntityGroup, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, OLD_ODEelement, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, StatisticsCollector, SourceModelComponent, SinkModelComponent, ModelElement, ModelComponent, EntityType, Entity, Counter, and Attribute.

7.97.3.2 _saveInstance() `virtual std::map<std::string, std::string>* PersistentObject_base::↩ _saveInstance ( ) [protected], [pure virtual]`

Implemented in Variable, Storage, Station, Set, Sequence, SequenceStep, Schedule, Resource, Queue, File, Failure, EntityGroup, Write, Unstore, Submodel, Store, Stop, Start, Signal, Separate, Seize, Search, Route, Remove, Release, Record, Process, PickUp, PickStation, OLD_ODEelement, Match, MarkovChain, LSODE, Leave, Hold, Exit, Enter, Dummy, DropOff, Dispose, Delay, Decide, Create, CelularAutomata, Batch, Assign, Access, StatisticsCollector, SourceModelComponent, SinkModelComponent, ModelElement, ModelComponent, EntityType, Entity, Counter, and Attribute.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PersistentObject_base.h

7.98 PickStation Class Reference

```
#include <PickStation.h>
```

Inheritance diagram for PickStation:

Collaboration diagram for PickStation:

Public Member Functions

- PickStation (Model ∗model, std::string name="")
- virtual ∼PickStation ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.98.1 Detailed Description

PickStation module DESCRIPTION The PickStation module allows an entity to select a particular station from the multiple stations specified. This module picks among the group of stations based on the selection logic defined with the module. The entity may then route, transport, convey, or connect to the station specified. If the method chosen is connect, the selected station is assigned to an entity attribute. The station selection process is based on the minimum or maximum value of a variety of system variables and expressions. TYPICAL USES A part sent to a processing station based on machine's availability at each station A loan application sent to a set of loan officers based on the number sent to each officer A customer selecting among cashier lines based on the least number waiting in each line PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Test Condition Test condition to use for the station selection process, either Minimum or Maximum. Number En Route to Station The number of entities transferring to the station is considered in the station selection process. Number in Queue The number of entities in the queue at the station is considered in the station selection process. Number of Resources Busy The number of busy resources at the station is considered in the station selection process. Expression Determines if an additional user-defined expression is considered in the station selection process. Transfer Type Determines how an entity will be transferred out of this module to its next destination station—either Route, Convey, Transport, or Connect. Save Attribute Defines the name of the attribute that will store the station name that is selected, visible when the transfer method is Connect. Route Time Move time of the entity from its current station to the station determined through this module. Units Time units for route-time parameters.

Definition at line 62 of file PickStation.h.

7.98.2 Constructor & Destructor Documentation

7.98.2.1 PickStation()

```
PickStation::PickStation (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file PickStation.cpp.

Here is the caller graph for this function:

7.98.2.2 ~PickStation()

```
virtual PickStation::~PickStation ( ) [virtual], [default]
```

7.98.3 Member Function Documentation

7.98.3.1 _check()

```
bool PickStation::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file PickStation.cpp.

7.98.3.2 _execute()

```
void PickStation::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file PickStation.cpp.

Here is the call graph for this function:

7.98.3.3 _initBetweenReplications()

```
void PickStation::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 48 of file PickStation.cpp.

7.98.3.4 _loadInstance()

```
bool PickStation::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 40 of file PickStation.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.98.3.5 _saveInstance()

```
std::map< std::string, std::string > * PickStation::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 51 of file PickStation.cpp.

Here is the call graph for this function:

7.98.3.6 GetPluginInformation() `PluginInformation * PickStation::GetPluginInformation ( ) [static]`

Definition at line 64 of file PickStation.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.98.3.7 LoadInstance() `ModelComponent * PickStation::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 25 of file PickStation.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.98.3.8 show() `std::string PickStation::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file PickStation.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/PickStation.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/PickStation.cpp

7.99 PickUp Class Reference

```
#include <PickUp.h>
```

Inheritance diagram for PickUp:

Collaboration diagram for PickUp:

Public Member Functions

- PickUp (Model *model, std::string name="")
- virtual ~PickUp ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelComponent * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)

Additional Inherited Members

7.99.1 Detailed Description

Pickup module DESCRIPTION The Pickup module removes a number of consecutive entities from a given queue starting at a specified rank in the queue. The entities that are picked up are added to the end of the incoming entity's group. TYPICAL USES Gathering an order from various queue locations Gathering completed forms for an office order Picking up students at a bus stop for school PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Quantity Number of entities to pick up. Queue Name Name of the queue from which the entities will be picked up, starting at the specified rank. Starting Rank Starting rank of the entities to pick up from the queue, Queue Name.

Definition at line 38 of file PickUp.h.

7.99.2 Constructor & Destructor Documentation

7.99.2.1 PickUp()

```
PickUp::PickUp (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file PickUp.cpp.

Here is the caller graph for this function:

7.99.2.2 ~PickUp()

```
virtual PickUp::~PickUp ( ) [virtual], [default]
```

7.99.3 Member Function Documentation

7.99.3.1 _check()

```
bool PickUp::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file PickUp.cpp.

7.99.3.2 _execute()

```
void PickUp::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file PickUp.cpp.

Here is the call graph for this function:

7.99.3.3 _initBetweenReplications() `void PickUp::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 48 of file PickUp.cpp.

7.99.3.4 _loadInstance() `bool PickUp::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 40 of file PickUp.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.99.3.5 _saveInstance() `std::map< std::string, std::string > * PickUp::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file PickUp.cpp.

Here is the call graph for this function:

7.99.3.6 GetPluginInformation() `PluginInformation * PickUp::GetPluginInformation ( ) [static]`

Definition at line 64 of file PickUp.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.99.3.7 LoadInstance()

```
ModelComponent * PickUp::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file PickUp.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.99.3.8 show()

```
std::string PickUp::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 21 of file PickUp.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/PickUp.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/PickUp.cpp

7.100 Plugin Class Reference

```
#include <Plugin.h>
```

Collaboration diagram for Plugin:

Public Member Functions

- Plugin (std::string filename_so_dll)
- Plugin (StaticGetPluginInformation getInformation)
- virtual ~Plugin ()=default
- bool isIsValidPlugin () const
- PluginInformation * getPluginInfo () const
- ModelElement * loadNew (Model *model, std::map< std::string, std::string > *fields)

  creates a new ModelElement from fields loaded from a file
- bool loadAndInsertNew (Model *model, std::map< std::string, std::string > *fields)

7.100.1 Detailed Description

A Plugin represents a dynamically linked component class (ModelComponent) or element class (ModelElement); It gives access to a ModelComponent so it can be used by the model. Classes like Create, Delay, and Dispose are examples of PlugIns. It corresponds directly to the "Expansible" part (the capitalized 'E') of the GenESyS acronymous PlugIns are NOT implemented yet

Definition at line 30 of file Plugin.h.

7.100.2 Constructor & Destructor Documentation

7.100.2.1 Plugin() [1/2]

```
Plugin::Plugin (
    std::string filename_so_dll )
```

7.100.2.2 Plugin() [2/2]

```
Plugin::Plugin (
    StaticGetPluginInformation getInformation )
```

Definition at line 23 of file Plugin.cpp.

7.100.2.3 ~Plugin()

```
virtual Plugin::~Plugin ( ) [virtual], [default]
```

7.100.3 Member Function Documentation

7.100.3.1 getPluginInfo()

```
PluginInformation * Plugin::getPluginInfo ( ) const
```

Definition at line 34 of file Plugin.cpp.

Here is the caller graph for this function:

7.100.3.2 isIsValidPlugin()

```
bool Plugin::isIsValidPlugin ( ) const
```

Definition at line 38 of file Plugin.cpp.

7.100.3.3 loadAndInsertNew()

```
bool Plugin::loadAndInsertNew (
    Model * model,
    std::map< std::string, std::string > * fields )
```

Definition at line 56 of file Plugin.cpp.

Here is the call graph for this function:

7.100.3.4 loadNew()

```
ModelElement * Plugin::loadNew (
    Model * model,
    std::map< std::string, std::string > * fields )
```

creates a new ModelElement from fields loaded from a file

Definition at line 48 of file Plugin.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Plugin.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Plugin.cpp

7.101 PluginConnector_if Class Reference

```
#include <PluginConnector_if.h>
```

Inheritance diagram for PluginConnector_if:

Collaboration diagram for PluginConnector_if:

Public Member Functions

- virtual Plugin ∗ check (const std::string dynamicLibraryFilename)=0
- virtual Plugin ∗ connect (const std::string dynamicLibraryFilename)=0
- virtual bool disconnect (const std::string dynamicLibraryFilename)=0
- virtual bool disconnect (Plugin ∗plugin)=0

7.101.1 Detailed Description

Definition at line 22 of file PluginConnector_if.h.

7.101.2 Member Function Documentation

7.101.2.1 check()

```
virtual Plugin* PluginConnector_if::check (
    const std::string dynamicLibraryFilename ) [pure virtual]
```

Implemented in PluginConnectorDummyImpl1.

Here is the caller graph for this function:

7.101.2.2 connect()

```
virtual Plugin* PluginConnector_if::connect (
    const std::string dynamicLibraryFilename ) [pure virtual]
```

Implemented in PluginConnectorDummyImpl1.

Here is the caller graph for this function:

7.101.2.3 disconnect() [1/2]

```
virtual bool PluginConnector_if::disconnect (
    const std::string dynamicLibraryFilename ) [pure virtual]
```

Implemented in PluginConnectorDummyImpl1.

Here is the caller graph for this function:

7.101.2.4 disconnect() [2/2]

```
virtual bool PluginConnector_if::disconnect (
    Plugin * plugin ) [pure virtual]
```

Implemented in PluginConnectorDummyImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginConnector_if.h

7.102 PluginConnectorDummyImpl1 Class Reference

```
#include <PluginConnectorDummyImpl1.h>
```

Inheritance diagram for PluginConnectorDummyImpl1:

Collaboration diagram for PluginConnectorDummyImpl1:

Public Member Functions

- PluginConnectorDummyImpl1 ()
- virtual ~PluginConnectorDummyImpl1 ()=default
- virtual Plugin * check (const std::string dynamicLibraryFilename)
- virtual Plugin * connect (const std::string dynamicLibraryFilename)
- virtual bool disconnect (const std::string dynamicLibraryFilename)
- virtual bool disconnect (Plugin *plugin)

7.102.1 Detailed Description

Definition at line 20 of file PluginConnectorDummyImpl1.h.

7.102.2 Constructor & Destructor Documentation

7.102.2.1 PluginConnectorDummyImpl1() `PluginConnectorDummyImpl1::PluginConnectorDummyImpl1 ( )`

Definition at line 74 of file PluginConnectorDummyImpl1.cpp.

7.102.2.2 ~PluginConnectorDummyImpl1() `virtual PluginConnectorDummyImpl1::~PluginConnector↩ DummyImpl1 ( ) [virtual], [default]`

7.102.3 Member Function Documentation

7.102.3.1 check() `Plugin * PluginConnectorDummyImpl1::check ( const std::string dynamicLibraryFilename ) [virtual]`

@

Todo :To implement

Implements PluginConnector_if.

Definition at line 77 of file PluginConnectorDummyImpl1.cpp.

7.102.3.2 connect() `Plugin * PluginConnectorDummyImpl1::connect ( const std::string dynamicLibraryFilename ) [virtual]`

Implements PluginConnector_if.

Definition at line 83 of file PluginConnectorDummyImpl1.cpp.

7.102.3.3 disconnect() [1/2] `bool PluginConnectorDummyImpl1::disconnect ( const std::string dynamicLibraryFilename ) [virtual]`

Implements PluginConnector_if.

Definition at line 202 of file PluginConnectorDummyImpl1.cpp.

7.102.3.4 disconnect() [2/2] `bool PluginConnectorDummyImpl1::disconnect ( Plugin * plugin ) [virtual]`

Implements PluginConnector_if.

Definition at line 208 of file PluginConnectorDummyImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.cpp

7.103 PluginInformation Class Reference

```
#include <PluginInformation.h>
```

Collaboration diagram for PluginInformation:

Public Member Functions

- PluginInformation (std::string pluginTypename, StaticLoaderComponentInstance componentloader)
- PluginInformation (std::string pluginTypename, StaticLoaderElementInstance elementloader)
- StaticLoaderElementInstance getElementLoader () const
- StaticLoaderComponentInstance GetComponentLoader () const
- bool isGenerateReport () const
- bool isComponent () const
- bool isSendTransfer () const
- bool isReceiveTransfer () const
- bool isSink () const
- bool isSource () const
- std::string getObservation () const
- std::string getVersion () const
- std::string getDate () const
- std::string getAuthor () const
- std::string getPluginTypename () const
- void insertDynamicLibFileDependence (std::string filename)
- void setDynamicLibFilenameDependencies (std::list< std::string > *dynamicLibFilenameDependencies)
- std::list< std::string > * getDynamicLibFilenameDependencies () const
- void setGenerateReport (bool generateReport)
- void setSendTransfer (bool sendTransfer)
- void setReceiveTransfer (bool receiveTransfer)
- void setSink (bool Sink)
- void setSource (bool Source)

- void setObservation (std::string observation)
- void setVersion (std::string version)
- void setDate (std::string date)
- void setAuthor (std::string author)
- void setMaximumOutputs (unsigned short _maximumOutputs)
- unsigned short getMaximumOutputs () const
- void setMinimumOutputs (unsigned short _minimumOutputs)
- unsigned short getMinimumOutputs () const
- void setMaximumInputs (unsigned short _maximumInputs)
- unsigned short getMaximumInputs () const
- void setMinimumInputs (unsigned short _minimumInputs)
- unsigned short getMinimumInputs () const
- void setDescriptionHelp (std::string _descriptionHelp)
- std::string getDescriptionHelp () const
- void setFields (std::map< std::string, std::string > *_fiewlds)
- std::map< std::string, std::string > * getFields () const

7.103.1 Detailed Description

Definition at line 33 of file PluginInformation.h.

7.103.2 Constructor & Destructor Documentation

7.103.2.1 PluginInformation() [1/2]

```
PluginInformation::PluginInformation (
    std::string pluginTypename,
    StaticLoaderComponentInstance componentloader )
```

Todo :

Definition at line 18 of file PluginInformation.cpp.

7.103.2.2 PluginInformation() [2/2]

```
PluginInformation::PluginInformation (
    std::string pluginTypename,
    StaticLoaderElementInstance elementloader )
```

Definition at line 25 of file PluginInformation.cpp.

7.103.3 Member Function Documentation

7.103.3.1 getAuthor() `std::string PluginInformation::getAuthor ( ) const`

Definition at line 76 of file PluginInformation.cpp.

7.103.3.2 GetComponentLoader() `StaticLoaderComponentInstance PluginInformation::GetComponent↩ Loader ( ) const`

Definition at line 36 of file PluginInformation.cpp.

7.103.3.3 getDate() `std::string PluginInformation::getDate ( ) const`

Definition at line 72 of file PluginInformation.cpp.

7.103.3.4 getDescriptionHelp() `std::string PluginInformation::getDescriptionHelp ( ) const`

Definition at line 168 of file PluginInformation.cpp.

7.103.3.5 getDynamicLibFilenameDependencies() `std::list< std::string > * PluginInformation↩ ::getDynamicLibFilenameDependencies ( ) const`

Definition at line 92 of file PluginInformation.cpp.

7.103.3.6 getElementLoader() `StaticLoaderElementInstance PluginInformation::getElementLoader ( ) const`

Definition at line 32 of file PluginInformation.cpp.

7.103.3.7 getFields() `std::map< std::string, std::string > * PluginInformation::getFields ( ) const`

Definition at line 176 of file PluginInformation.cpp.

7.103.3.8 getMaximumInputs() `unsigned short PluginInformation::getMaximumInputs ( ) const`

Definition at line 152 of file PluginInformation.cpp.

7.103.3.9 getMaximumOutputs() `unsigned short PluginInformation::getMaximumOutputs ( ) const`

Definition at line 136 of file PluginInformation.cpp.

7.103.3.10 getMinimumInputs() `unsigned short PluginInformation::getMinimumInputs ( ) const`

Definition at line 160 of file PluginInformation.cpp.

7.103.3.11 getMinimumOutputs() `unsigned short PluginInformation::getMinimumOutputs ( ) const`

Definition at line 144 of file PluginInformation.cpp.

7.103.3.12 getObservation() `std::string PluginInformation::getObservation ( ) const`

Definition at line 64 of file PluginInformation.cpp.

7.103.3.13 getPluginTypename() `std::string PluginInformation::getPluginTypename ( ) const`

Definition at line 80 of file PluginInformation.cpp.

7.103.3.14 getVersion() `std::string PluginInformation::getVersion ( ) const`

Definition at line 68 of file PluginInformation.cpp.

7.103.3.15 insertDynamicLibFileDependence()

```
void PluginInformation::insertDynamicLibFile↩
Dependence (
            std::string filename )
```

Definition at line 84 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.16 isComponent() `bool PluginInformation::isComponent ( ) const`

Definition at line 44 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.17 isGenerateReport() `bool PluginInformation::isGenerateReport ( ) const`

Definition at line 40 of file PluginInformation.cpp.

7.103.3.18 isReceiveTransfer() `bool PluginInformation::isReceiveTransfer ( ) const`

Definition at line 52 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.19 isSendTransfer() `bool PluginInformation::isSendTransfer ( ) const`

Definition at line 48 of file PluginInformation.cpp.

7.103.3.20 isSink() `bool PluginInformation::isSink ( ) const`

Definition at line 56 of file PluginInformation.cpp.

7.103.3.21 isSource() `bool PluginInformation::isSource ( ) const`

Definition at line 60 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.22 setAuthor() `void PluginInformation::setAuthor ( std::string author )`

Definition at line 128 of file PluginInformation.cpp.

7.103.3.23 setDate() `void PluginInformation::setDate ( std::string date )`

Definition at line 124 of file PluginInformation.cpp.

7.103.3.24 setDescriptionHelp() `void PluginInformation::setDescriptionHelp ( std::string _descriptionHelp )`

Definition at line 164 of file PluginInformation.cpp.

7.103.3.25 setDynamicLibFilenameDependencies() `void PluginInformation::setDynamicLibFilenameDependencies ( std::list< std::string > * dynamicLibFilenameDependencies )`

Definition at line 88 of file PluginInformation.cpp.

7.103.3.26 setFields() `void PluginInformation::setFields ( std::map< std::string, std::string > * _fiewlds )`

Definition at line 172 of file PluginInformation.cpp.

7.103.3.27 setGenerateReport()

```
void PluginInformation::setGenerateReport (
    bool generateReport )
```

Definition at line 96 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.28 setMaximumInputs()

```
void PluginInformation::setMaximumInputs (
    unsigned short _maximumInputs )
```

Definition at line 148 of file PluginInformation.cpp.

7.103.3.29 setMaximumOutputs()

```
void PluginInformation::setMaximumOutputs (
    unsigned short _maximumOutputs )
```

Definition at line 132 of file PluginInformation.cpp.

7.103.3.30 setMinimumInputs()

```
void PluginInformation::setMinimumInputs (
    unsigned short _minimumInputs )
```

Definition at line 156 of file PluginInformation.cpp.

7.103.3.31 setMinimumOutputs()

```
void PluginInformation::setMinimumOutputs (
    unsigned short _minimumOutputs )
```

Definition at line 140 of file PluginInformation.cpp.

7.103.3.32 setObservation()

```
void PluginInformation::setObservation (
    std::string observation )
```

Definition at line 116 of file PluginInformation.cpp.

7.103.3.33 setReceiveTransfer() `void PluginInformation::setReceiveTransfer ( bool receiveTransfer )`

Definition at line 104 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.34 setSendTransfer() `void PluginInformation::setSendTransfer ( bool sendTransfer )`

Definition at line 100 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.35 setSink() `void PluginInformation::setSink ( bool Sink )`

Definition at line 108 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.36 setSource() `void PluginInformation::setSource ( bool Source )`

Definition at line 112 of file PluginInformation.cpp.

Here is the caller graph for this function:

7.103.3.37 setVersion() `void PluginInformation::setVersion ( std::string version )`

Definition at line 120 of file PluginInformation.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginInformation.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginInformation.cpp

7.104 PluginManager Class Reference

```
#include <PluginManager.h>
```

Collaboration diagram for PluginManager:

Public Member Functions

- PluginManager (Simulator *simulator)
- virtual ~PluginManager ()=default
- bool check (const std::string dynamicLibraryFilename)
- Plugin * insert (const std::string dynamicLibraryFilename)
- bool remove (const std::string dynamicLibraryFilename)
- bool remove (Plugin *plugin)
- Plugin * find (std::string pluginTypeName)
- Plugin * front ()
- Plugin * next ()
- Plugin * last ()

7.104.1 Detailed Description

Definition at line 24 of file PluginManager.h.

7.104.2 Constructor & Destructor Documentation

7.104.2.1 PluginManager() `PluginManager::PluginManager ( Simulator * simulator )`

Definition at line 24 of file PluginManager.cpp.

7.104.2.2 ~PluginManager() `virtual PluginManager::~PluginManager ( ) [virtual], [default]`

7.104.3 Member Function Documentation

7.104.3.1 check() `bool PluginManager::check ( const std::string dynamicLibraryFilename )`

Definition at line 96 of file PluginManager.cpp.

Here is the call graph for this function:

7.104.3.2 find()

```
Plugin * PluginManager::find (
    std::string pluginTypeName )
```

Definition at line 143 of file PluginManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.104.3.3 front()

```
Plugin * PluginManager::front ( )
```

Definition at line 153 of file PluginManager.cpp.

Here is the call graph for this function:

7.104.3.4 insert()

```
Plugin * PluginManager::insert (
    const std::string dynamicLibraryFilename )
```

Definition at line 106 of file PluginManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.104.3.5 last() `Plugin * PluginManager::last ( )`

Definition at line 163 of file PluginManager.cpp.

Here is the call graph for this function:

7.104.3.6 next() `Plugin * PluginManager::next ( )`

Definition at line 158 of file PluginManager.cpp.

Here is the call graph for this function:

7.104.3.7 remove() [1/2] `bool PluginManager::remove ( const std::string dynamicLibraryFilename )`

Definition at line 122 of file PluginManager.cpp.

Here is the call graph for this function:

7.104.3.8 remove() [2/2] bool PluginManager::remove (Plugin * plugin)

Definition at line 128 of file PluginManager.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/PluginManager.cpp

7.105 yy::position Class Reference

A point in a source file.

```
#include <location.hh>
```

Collaboration diagram for yy::position:

Public Types

- typedef const std::string filename_type

  Type for file name.
- typedef int counter_type

  Type for line and column numbers.

Public Member Functions

- position (filename_type ∗f=YY_NULLPTR, counter_type l=1, counter_type c=1)

  Construct a position.
- void initialize (filename_type ∗fn=YY_NULLPTR, counter_type l=1, counter_type c=1)

  Initialization.

Line and Column related manipulators

- void lines (counter_type count=1)

  (line related) Advance to the COUNT next lines.
- void columns (counter_type count=1)

  (column related) Advance to the COUNT next columns.

Public Attributes

- filename_type * filename

  File name to which this position refers.
- counter_type line

  Current line number.
- counter_type column

  Current column number.

7.105.1 Detailed Description

A point in a source file.

Definition at line 60 of file location.hh.

7.105.2 Member Typedef Documentation

7.105.2.1 counter_type `typedef int yy::position::counter_type`

Type for line and column numbers.

Definition at line 66 of file location.hh.

7.105.2.2 filename_type `typedef const std::string yy::position::filename_type`

Type for file name.

Definition at line 64 of file location.hh.

7.105.3 Constructor & Destructor Documentation

7.105.3.1 position()

```
yy::position::position (
    filename_type * f = YY_NULLPTR,
    counter_type l = 1,
    counter_type c = 1 )  [inline], [explicit]
```

Construct a position.

Definition at line 69 of file location.hh.

7.105.4 Member Function Documentation

7.105.4.1 columns()

```
void yy::position::columns (
    counter_type count = 1 ) [inline]
```

(column related) Advance to the COUNT next columns.

Definition at line 101 of file location.hh.

Here is the caller graph for this function:

7.105.4.2 initialize()

```
void yy::position::initialize (
    filename_type * fn = YY_NULLPTR,
    counter_type l = 1,
    counter_type c = 1 ) [inline]
```

Initialization.

Definition at line 79 of file location.hh.

Here is the caller graph for this function:

7.105.4.3 lines()

```
void yy::position::lines (
    counter_type count = 1 ) [inline]
```

(line related) Advance to the COUNT next lines.

Definition at line 91 of file location.hh.

Here is the caller graph for this function:

7.105.5 Member Data Documentation

7.105.5.1 column

```
counter_type yy::position::column
```

Current column number.

Definition at line 112 of file location.hh.

7.105.5.2 filename

```
filename_type* yy::position::filename
```

File name to which this position refers.

Definition at line 108 of file location.hh.

7.105.5.3 line

```
counter_type yy::position::line
```

Current line number.

Definition at line 110 of file location.hh.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/location.hh

7.106 ProbabilityDistribution Class Reference

```
#include <ProbabilityDistribution.h>
```

Collaboration diagram for ProbabilityDistribution:

Static Public Member Functions

- static double beta (double x, double alpha, double beta)
- static double chi2 (double x, double degreeFreedom)
- static double erlang (double x, double shape, double scale)
- static double exponential (double x, double mean)
- static double fisherSnedecor (double x, double d1, double d2)
- static double gamma (double x, double shape, double scale)
- static double logNormal (double x, double mean, double stddev)
- static double normal (double x, double mean, double stddev)
- static double poisson (double x, double mean)
- static double triangular (double x, double min, double mode, double max)
- static double tStudent (double x, double mean, double stddev, double degreeFreedom)
- static double uniform (double x, double min, double max)
- static double weibull (double x, double shape, double scale)
- static double inverseChi2 (double cumulativeProbability, double degreeFreedom)
- static double inverseFFisherSnedecor (double cumulativeProbability, double d1, double d2)
- static double inverseNormal (double cumulativeProbability, double mean, double stddev)
- static double inverseTStudent (double cumulativeProbability, double mean, double stddev, double degree↩ Freedom)

7.106.1 Detailed Description

Definition at line 20 of file ProbabilityDistribution.h.

7.106.2 Member Function Documentation

7.106.2.1 beta()

```
double ProbabilityDistribution::beta (
    double x,
    double alpha,
    double beta ) [static]
```

Definition at line 21 of file ProbabilityDistribution.cpp.

Here is the caller graph for this function:

7.106.2.2 chi2()

```
double ProbabilityDistribution::chi2 (
    double x,
    double degreeFreedom ) [static]
```

Definition at line 28 of file ProbabilityDistribution.cpp.

Here is the caller graph for this function:

7.106.2.3 erlang()

```
double ProbabilityDistribution::erlang (
    double x,
    double shape,
    double scale ) [static]
```

Definition at line 37 of file ProbabilityDistribution.cpp.

7.106.2.4 exponential()

```
double ProbabilityDistribution::exponential (
    double x,
    double mean ) [static]
```

Definition at line 44 of file ProbabilityDistribution.cpp.

7.106.2.5 fisherSnedecor()

```
double ProbabilityDistribution::fisherSnedecor (
    double x,
    double d1,
    double d2 ) [static]
```

Definition at line 51 of file ProbabilityDistribution.cpp.

Here is the caller graph for this function:

7.106.2.6 gamma()

```
double ProbabilityDistribution::gamma (
    double x,
    double shape,
    double scale ) [static]
```

Definition at line 59 of file ProbabilityDistribution.cpp.

7.106.2.7 inverseChi2()

```
double ProbabilityDistribution::inverseChi2 (
    double cumulativeProbability,
    double degreeFreedom ) [static]
```

TODO: Is there a way to better determine the upper limit?

Definition at line 129 of file ProbabilityDistribution.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.106.2.8 inverseFFisherSnedecor()

```
double ProbabilityDistribution::inverseFFisherSnedecor (
    double cumulativeProbability,
    double d1,
    double d2 ) [static]
```

TODO: Is there a way to better determine the upper limit?

Definition at line 165 of file ProbabilityDistribution.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.106.2.9 inverseNormal()

```
double ProbabilityDistribution::inverseNormal (
    double cumulativeProbability,
    double mean,
    double stddev ) [static]
```

TODO: Could be better (separate into 1,2,3xstddev)

Definition at line 201 of file ProbabilityDistribution.cpp.

Here is the caller graph for this function:

7.106.2.10 inverseTStudent()

```
double ProbabilityDistribution::inverseTStudent (
    double cumulativeProbability,
    double mean,
    double stddev,
    double degreeFreedom ) [static]
```

TODO: Could be better

Definition at line 240 of file ProbabilityDistribution.cpp.

Here is the caller graph for this function:

7.106.2.11 logNormal()

```
double ProbabilityDistribution::logNormal (
    double x,
    double mean,
    double stddev ) [static]
```

Definition at line 65 of file ProbabilityDistribution.cpp.

7.106.2.12 normal()

```
double ProbabilityDistribution::normal (
    double x,
    double mean,
    double stddev ) [static]
```

Definition at line 71 of file ProbabilityDistribution.cpp.

7.106.2.13 poisson()

```
double ProbabilityDistribution::poisson (
    double x,
    double mean ) [static]
```

Definition at line 77 of file ProbabilityDistribution.cpp.

7.106.2.14 triangular()

```
double ProbabilityDistribution::triangular (
    double x,
    double min,
    double mode,
    double max ) [static]
```

Definition at line 84 of file ProbabilityDistribution.cpp.

7.106.2.15 tStudent()

```
double ProbabilityDistribution::tStudent (
    double x,
    double mean,
    double stddev,
    double degreeFreedom ) [static]
```

Definition at line 95 of file ProbabilityDistribution.cpp.

Here is the caller graph for this function:

7.106.2.16 uniform()

```
double ProbabilityDistribution::uniform (
    double x,
    double min,
    double max ) [static]
```

Definition at line 101 of file ProbabilityDistribution.cpp.

7.106.2.17 weibull()

```
double ProbabilityDistribution::weibull (
    double x,
    double shape,
    double scale ) [static]
```

Definition at line 105 of file ProbabilityDistribution.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/ProbabilityDistribution.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/ProbabilityDistribution.cpp

7.107 Process Class Reference

```
#include <Process.h>
```

Inheritance diagram for Process:

Collaboration diagram for Process:

Public Member Functions

- Process (Model ∗model, std::string name="")
- virtual ∼Process ()=default
- virtual std::string show ()
- void setPriority (unsigned short _priority)
- unsigned short getPriority () const
- void setAllocationType (unsigned int _allocationType)
- unsigned int getAllocationType () const
- List< SeizableItem ∗ > ∗ getSeizeRequests () const
- void setQueueableItem (QueueableItem ∗_queueableItem)
- QueueableItem ∗ getQueueableItem () const
- void setSaveAttribute (std::string _saveAttribute)
- std::string getSaveAttribute () const
- void setDelayExpression (std::string _delayExpression)
- std::string delayExpression () const
- void setDelayTimeUnit (Util::TimeUnit _delayTimeUnit)
- Util::TimeUnit delayTimeUnit () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.107.1 Detailed Description

This component ...

Definition at line 25 of file Process.h.

7.107.2 Constructor & Destructor Documentation

7.107.2.1 Process()

```
Process::Process (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.2.2 ~Process() `virtual Process::~Process ( ) [virtual], [default]`

7.107.3 Member Function Documentation

7.107.3.1 _check() `bool Process::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 127 of file Process.cpp.

Here is the call graph for this function:

7.107.3.2 _createInternalElements() `void Process::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 99 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.3 _execute()

```
void Process::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 87 of file Process.cpp.

Here is the call graph for this function:

7.107.3.4 _loadInstance()

```
bool Process::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 91 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.5 _saveInstance() `std::map< std::string, std::string > * Process::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 121 of file Process.cpp.

Here is the call graph for this function:

7.107.3.6 delayExpression() `std::string Process::delayExpression ( ) const`

Definition at line 65 of file Process.cpp.

Here is the call graph for this function:

7.107.3.7 delayTimeUnit() `Util::TimeUnit Process::delayTimeUnit ( ) const`

Definition at line 73 of file Process.cpp.

Here is the call graph for this function:

7.107.3.8 getAllocationType() `unsigned int Process::getAllocationType ( ) const`

Definition at line 37 of file Process.cpp.

Here is the call graph for this function:

7.107.3.9 GetPluginInformation() `PluginInformation * Process::GetPluginInformation ( ) [static]`

Definition at line 141 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.10 getPriority() `unsigned short Process::getPriority ( ) const`

Definition at line 29 of file Process.cpp.

Here is the call graph for this function:

7.107.3.11 getQueueableItem() `QueueableItem * Process::getQueueableItem ( ) const`

Definition at line 49 of file Process.cpp.

Here is the call graph for this function:

7.107.3.12 getSaveAttribute() `std::string Process::getSaveAttribute ( ) const`

Definition at line 57 of file Process.cpp.

Here is the call graph for this function:

7.107.3.13 getSeizeRequests() `List< SeizableItem * > * Process::getSeizeRequests ( ) const`

Definition at line 41 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.14 LoadInstance() `ModelComponent * Process::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 77 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.15 setAllocationType() `void Process::setAllocationType ( unsigned int _allocationType )`

Definition at line 33 of file Process.cpp.

Here is the call graph for this function:

7.107.3.16 setDelayExpression() `void Process::setDelayExpression ( std::string _delayExpression )`

Definition at line 61 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.17 setDelayTimeUnit()

```
void Process::setDelayTimeUnit (
    Util::TimeUnit _delayTimeUnit )
```

Definition at line 69 of file Process.cpp.

Here is the call graph for this function:

7.107.3.18 setPriority()

```
void Process::setPriority (
    unsigned short _priority )
```

Definition at line 25 of file Process.cpp.

Here is the call graph for this function:

7.107.3.19 setQueueableItem()

```
void Process::setQueueableItem (
    QueueableItem * _queueableItem )
```

Definition at line 45 of file Process.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.107.3.20 setSaveAttribute() `void Process::setSaveAttribute ( std::string _saveAttribute )`

Definition at line 53 of file Process.cpp.

Here is the call graph for this function:

7.107.3.21 show() `std::string Process::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file Process.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Process.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Process.cpp

7.108 QMainWindow Class Reference

Inheritance diagram for QMainWindow:

Collaboration diagram for QMainWindow:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mainWindow/mainwindow.h

7.109 Queue Class Reference

```
#include <Queue.h>
```

Inheritance diagram for Queue:

Collaboration diagram for Queue:

Public Types

- enum class OrderRule : int { FIFO = 1 , LIFO = 2 , HIGHESTVALUE = 3 , SMALLESTVALUE = 4 }

Public Member Functions

- Queue (Model *model, std::string name="")
- virtual ~Queue ()
- virtual std::string show ()
- void insertElement (Waiting *element)
- void removeElement (Waiting *element)
- unsigned int size ()
- Waiting * first ()
- Waiting * getAtRank (unsigned int rank)
- void setAttributeName (std::string _attributeName)
- std::string getAttributeName () const
- void setOrderRule (OrderRule _orderRule)
- Queue::OrderRule getOrderRule () const
- double sumAttributesFromWaiting (Util::identification attributeID)
- double getAttributeFromWaitingRank (unsigned int rank, Util::identification attributeID)
- void initBetweenReplications ()

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _createInternalElements ()
- virtual ParserChangesInformation * _getParserChangesInformation ()

Additional Inherited Members

7.109.1 Detailed Description

Queue module DESCRIPTION This data module may be utilized to change the ranking rule for a specified queue. The default ranking rule for all queues is First In, First Out unless otherwise specified in this module. There is an additional field that allows the queue to be defined as shared. TYPICAL USES Stack of work waiting for a resource at a Process module Holding area for documents waiting to be collated at a Batch module Prompt Description Name The name of the queue whose characteristics are being defined. This name must be unique. Type Ranking rule for the queue, which can be based on an attribute. Types include First In, First Out; Last In, First Out; Lowest Attribute Value (first); and Highest Attribute Value (first). A low attribute value would be 0 or 1, while a high value may be 200 or 300. Attribute Name Attribute that will be evaluated for the Lowest Attribute Value or Highest Attribute Value types. Entities with lowest or highest values of the attribute will be ranked first in the queue, with ties being broken using the First In, First Out rule. Shared Check box that determines whether a specific queue is used in multiple places within the simulation model. Shared queues can only be used for seizing resources (for example, with the Seize module from the Advanced Process panel). Report Statistics Specifies whether or not statistics will be collected automatically and stored in the report database for this queue.

Definition at line 91 of file Queue.h.

7.109.2 Member Enumeration Documentation

7.109.2.1 OrderRule `enum Queue::OrderRule : int [strong]`

Enumerator

FIFO	
LIFO	
HIGHESTVALUE	
SMALLESTVALUE	

Definition at line 94 of file Queue.h.

7.109.3 Constructor & Destructor Documentation

7.109.3.1 Queue()

```
Queue::Queue (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Queue.cpp.

Here is the caller graph for this function:

7.109.3.2 ~Queue()

```
Queue::~Queue ( ) [virtual]
```

Definition at line 21 of file Queue.cpp.

7.109.4 Member Function Documentation

7.109.4.1 _check()

```
bool Queue::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 129 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.2 _createInternalElements() `void Queue::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist before checking such expression

Reimplemented from ModelElement.

Definition at line 133 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.3 _getParserChangesInformation() `ParserChangesInformation * Queue::_getParserChanges↩ Information ( ) [protected], [virtual]`

This method returns all changes in the parser that are needed by plugins of this ModelElements. When connecting a new plugin, ParserChangesInformation are used to change parser source code, whch is after compiled and dinamically linked to to simulator kernel to reflect the changes

Reimplemented from ModelElement.

Definition at line 151 of file Queue.cpp.

7.109.4.4 _loadInstance() `bool Queue::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 110 of file Queue.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.109.4.5 _saveInstance() `std::map< std::string, std::string > * Queue::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 122 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.6 first() `Waiting * Queue::first ( )`

Definition at line 55 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.7 getAtRank()

```
Waiting * Queue::getAtRank (
    unsigned int rank )
```

Definition at line 59 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.8 getAttributeFromWaitingRank()

```
double Queue::getAttributeFromWaitingRank (
    unsigned int rank,
    Util::identification attributeID )
```

Definition at line 87 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.9 getAttributeName()

```
std::string Queue::getAttributeName ( ) const
```

Definition at line 67 of file Queue.cpp.

7.109.4.10 getOrderRule()

```
Queue::OrderRule Queue::getOrderRule ( ) const
```

Definition at line 75 of file Queue.cpp.

7.109.4.11 GetPluginInformation() `PluginInformation * Queue::GetPluginInformation ( ) [static]`

Definition at line 95 of file Queue.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.109.4.12 initBetweenReplications() `void Queue::initBetweenReplications ( )`

Definition at line 47 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.13 insertElement() `void Queue::insertElement ( Waiting * element )`

Definition at line 31 of file Queue.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.109.4.14 LoadInstance()

```
ModelElement * Queue::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 100 of file Queue.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.109.4.15 removeElement() `void Queue::removeElement ( Waiting * element )`

Definition at line 37 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.16 setAttributeName() `void Queue::setAttributeName ( std::string _attributeName )`

Definition at line 63 of file Queue.cpp.

7.109.4.17 setOrderRule() `void Queue::setOrderRule ( OrderRule _orderRule )`

Definition at line 71 of file Queue.cpp.

Here is the caller graph for this function:

7.109.4.18 show() `std::string Queue::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 26 of file Queue.cpp.

Here is the call graph for this function:

7.109.4.19 size() `unsigned int Queue::size ( )`

Definition at line 51 of file Queue.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.109.4.20 sumAttributesFromWaiting() `double Queue::sumAttributesFromWaiting ( Util::identification attributeID )`

Definition at line 79 of file Queue.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Queue.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Queue.cpp

7.110 QueueableItem Class Reference

```
#include <QueueableItem.h>
```

Collaboration diagram for QueueableItem:

Public Types

- enum class QueueableType : int { QUEUE = 1 , SET = 2 }

Public Member Functions

- QueueableItem (ModelElement *queueOrSet, QueueableItem::QueueableType queueableType=QueueableItem::QueueableTyp
  std::string index="0")
- bool loadInstance (std::map< std::string, std::string > *fields)
- std::map< std::string, std::string > * saveInstance ()
- std::string show ()
- void setIndex (std::string index)
- std::string getIndex () const
- std::string getQueueableName () const
- void setQueue (Queue *resource)
- Queue * getQueue () const
- void setSet (Set *set)
- Set * getSet () const
- void setQueueableType (QueueableType queueableType)
- QueueableType getQueueableType () const
- ModelElement * getQueueable () const
- void setElementManager (ElementManager *_elementManager)

7.110.1 Detailed Description

Definition at line 23 of file QueueableItem.h.

7.110.2 Member Enumeration Documentation

7.110.2.1 QueueableType `enum QueueableItem::QueueableType : int [strong]`

Enumerator

QUEUE	
SET	

Definition at line 26 of file QueueableItem.h.

7.110.3 Constructor & Destructor Documentation

7.110.3.1 QueueableItem()

```
QueueableItem::QueueableItem (
    ModelElement * queueOrSet,
    QueueableItem::QueueableType queueableType = QueueableItem::QueueableType::QUEUE,
    std::string index = "0" )
```

Definition at line 19 of file QueueableItem.cpp.

7.110.4 Member Function Documentation

7.110.4.1 getIndex() `std::string QueueableItem::getIndex ( ) const`

Definition at line 98 of file QueueableItem.cpp.

Here is the caller graph for this function:

7.110.4.2 getQueue() `Queue * QueueableItem::getQueue ( ) const`

Definition at line 111 of file QueueableItem.cpp.

Here is the caller graph for this function:

7.110.4.3 getQueueable() `ModelElement * QueueableItem::getQueueable ( ) const`

Definition at line 132 of file QueueableItem.cpp.

7.110.4.4 getQueueableName() `std::string QueueableItem::getQueueableName ( ) const`

Definition at line 102 of file QueueableItem.cpp.

7.110.4.5 getQueueableType() `QueueableItem::QueueableType QueueableItem::getQueueableType ( ) const`

Definition at line 128 of file QueueableItem.cpp.

Here is the caller graph for this function:

7.110.4.6 getSet() `Set * QueueableItem::getSet ( ) const`

Definition at line 120 of file QueueableItem.cpp.

Here is the caller graph for this function:

7.110.4.7 loadInstance() `bool QueueableItem::loadInstance ( std::map< std::string, std::string > * fields )`

Definition at line 33 of file QueueableItem.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.110.4.8 saveInstance() `std::map< std::string, std::string > * QueueableItem::saveInstance ( )`

Definition at line 55 of file QueueableItem.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.110.4.9 setElementManager() `void QueueableItem::setElementManager ( ElementManager * _elementManager )`

Definition at line 136 of file QueueableItem.cpp.

Here is the caller graph for this function:

7.110.4.10 setIndex() `void QueueableItem::setIndex ( std::string index )`

Definition at line 94 of file QueueableItem.cpp.

7.110.4.11 setQueue()

```
void QueueableItem::setQueue (
    Queue * resource )
```

Definition at line 106 of file QueueableItem.cpp.

Here is the call graph for this function:

7.110.4.12 setQueueableType()

```
void QueueableItem::setQueueableType (
    QueueableItem::QueueableType queueableType )
```

Definition at line 124 of file QueueableItem.cpp.

7.110.4.13 setSet()

```
void QueueableItem::setSet (
    Set * set )
```

Definition at line 115 of file QueueableItem.cpp.

Here is the call graph for this function:

7.110.4.14 show() `std::string QueueableItem::show ( )`

Definition at line 90 of file QueueableItem.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/QueueableItem.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/QueueableItem.cpp

7.111 Record Class Reference

```
#include <Record.h>
```

Inheritance diagram for Record:

Collaboration diagram for Record:

Public Member Functions

- Record (Model ∗model, std::string name="")
- virtual ∼Record ()
- void setFilename (std::string filename)
- std::string getFilename () const
- void setExpression (const std::string expression)
- std::string getExpression () const
- void setExpressionName (std::string expressionName)
- std::string getExpressionName () const
- StatisticsCollector ∗ getCstatExpression () const
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.111.1 Detailed Description

Record module DESCRIPTION This module is used to collect statistics in the simulation model. Various types of observational statistics are available, including time between exits through the module, entity statistics (such as time or costing), general observations, and interval statistics (from some time stamp to the current simulation time). A count type of statistic is available as well. Tally and Counter sets can also be specified. TYPICAL USES Collect the number of jobs completed each hour Count how many orders have been late being fulfilled Record the time spent by priority customers in the main check-out line PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Type of observational (tally) or count statistic to be generated. Count will increase or decrease the value of the named statistic by the specified value. Entity Statistics will generate general entity statistics, such as time and costing/duration information. Time Interval will calculate and record the difference between a specified attribute's value and current simulation time. Time Between will track and record the time between entities entering the module. Expression will record the value of the specified expression. Attribute Name Name of the attribute whose value will be used for the interval statistics. Applies only when Type is Interval. Value Value that will be recorded to the observational statistic when Type is Expression or added to the counter when Type is Count. Tally Name This field defines the symbol name of the tally into which the observation is to be recorded. Applies only when Type is Time Interval, Time Between, or Expression. Counter This field defines the symbol name of the counter to Name increment/decrement. Applies only when Type is Counter. Record into Set Check box to specify whether or not a tally or counter set will be used. Tally Set Name Name of the tally set that will be used to record the observational-type statistic. Applies only when Type is Time Interval, Time Between, or Expression. Counter Set Name Name of the counter set that will be used to record the count-type statistic. Applies only when Type is Count. Set Index Index into the tally or counter set.

Definition at line 62 of file Record.h.

7.111.2 Constructor & Destructor Documentation

7.111.2.1 Record()

```
Record::Record (
    Model * model,
    std::string name = "" )
```

Definition at line 20 of file Record.cpp.

Here is the caller graph for this function:

7.111.2.2 ~Record() `Record::~Record ( ) [virtual]`

Definition at line 23 of file Record.cpp.

Here is the call graph for this function:

7.111.3 Member Function Documentation

7.111.3.1 _check() `bool Record::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 100 of file Record.cpp.

Here is the call graph for this function:

7.111.3.2 _createInternalElements() `void Record::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 106 of file Record.cpp.

Here is the call graph for this function:

7.111.3.3 _execute()

```
void Record::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 64 of file Record.cpp.

Here is the call graph for this function:

7.111.3.4 _initBetweenReplications()

```
void Record::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 97 of file Record.cpp.

7.111.3.5 _loadInstance()

`bool Record::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 87 of file Record.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.111.3.6 _saveInstance()

`std::map< std::string, std::string > * Record::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 79 of file Record.cpp.

Here is the call graph for this function:

7.111.3.7 getCstatExpression()

`StatisticsCollector * Record::getCstatExpression ( ) const`

Definition at line 44 of file Record.cpp.

7.111.3.8 getExpression() `std::string Record::getExpression ( ) const`

Definition at line 60 of file Record.cpp.

7.111.3.9 getExpressionName() `std::string Record::getExpressionName ( ) const`

Definition at line 40 of file Record.cpp.

7.111.3.10 getFilename() `std::string Record::getFilename ( ) const`

Definition at line 52 of file Record.cpp.

7.111.3.11 GetPluginInformation() `PluginInformation * Record::GetPluginInformation ( ) [static]`

Definition at line 117 of file Record.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.111.3.12 LoadInstance()

```
ModelComponent * Record::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 122 of file Record.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.111.3.13 setExpression()

```
void Record::setExpression (
    const std::string expression )
```

Definition at line 56 of file Record.cpp.

Here is the caller graph for this function:

7.111.3.14 setExpressionName() `void Record::setExpressionName ( std::string expressionName )`

Definition at line 34 of file Record.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.111.3.15 setFilename() `void Record::setFilename ( std::string filename )`

Definition at line 48 of file Record.cpp.

Here is the caller graph for this function:

7.111.3.16 show() `std::string Record::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 27 of file Record.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Record.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Record.cpp

7.112 Release Class Reference

```
#include <Release.h>
```

Inheritance diagram for Release:

Collaboration diagram for Release:

Public Member Functions

- Release (Model ∗model, std::string name="")
- virtual ∼Release ()=default
- virtual std::string show ()
- void setPriority (unsigned short _priority)
- unsigned short priority () const
- List< SeizableItem ∗ > ∗ getReleaseRequests () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.112.1 Detailed Description

Release module DESCRIPTION The Release module is used to release units of a resource that an entity previously has seized. This module may be used to release individual resources or may be used to release resources within a set. For each resource to be released, the name and quantity to release are specified. When the entity enters the Release module, it gives up control of the specified resource(s). Any entities waiting in queues for those resources will gain control of the resources immediately. TYPICAL USES Finishing a customer order (release the operator) Completing a tax return (release the accountant) Leaving the hospital (release the doctor, nurse, hospital room) PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Type of resource for releasing, either specifying a particular resource, or selecting from a pool of resources (that is, a resource set). The resource name may also be specified by an expression or attribute value. Resource Name Name of the resource that will be released. Set Name Name of the resource set from which a member will be released. Attribute Name Name of the attribute that specifies the resource name to be released. Expression Name of the expression that specifies the name of the resource to be released. Quantity Number of resources of a given name or from a given set that will be released. For sets, this value specifies only the number of a selected resource that will be released (based on the resource's capacity), not the number of members to be released within the set. Release Rule Method of determining which resource within a set to release. Last Member Seized and First Member Seized will release the last/first member from within the set that was seized. Specific member indicates that a member number or attribute (with a member number value) will be used to specify the member to release. Set Index Member index of the resource set that the entity will release.

Definition at line 64 of file Release.h.

7.112.2 Constructor & Destructor Documentation

7.112.2.1 Release()

```
Release::Release (
    Model * model,
    std::string name = "" )
```

Definition at line 21 of file Release.cpp.

Here is the caller graph for this function:

7.112.2.2 ~Release()

```
virtual Release::~Release ( ) [virtual], [default]
```

7.112.3 Member Function Documentation

7.112.3.1 _check()

```
bool Release::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 133 of file Release.cpp.

Here is the call graph for this function:

7.112.3.2 _execute()

```
void Release::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 56 of file Release.cpp.

Here is the call graph for this function:

7.112.3.3 _initBetweenReplications() `void Release::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 94 of file Release.cpp.

7.112.3.4 _loadInstance() `bool Release::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 103 of file Release.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.112.3.5 _saveInstance() `std::map< std::string, std::string > * Release::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 120 of file Release.cpp.

Here is the call graph for this function:

7.112.3.6 GetPluginInformation() `PluginInformation * Release::GetPluginInformation ( ) [static]`

Definition at line 161 of file Release.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.112.3.7 getReleaseRequests() `List< SeizableItem * > * Release::getReleaseRequests ( ) const`

Definition at line 44 of file Release.cpp.

Here is the caller graph for this function:

7.112.3.8 LoadInstance()

```
ModelComponent * Release::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields )  [static]
```

Definition at line 167 of file Release.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.112.3.9 priority()

```
unsigned short Release::priority ( ) const
```

Definition at line 40 of file Release.cpp.

7.112.3.10 setPriority()

```
void Release::setPriority (
    unsigned short _priority )
```

Definition at line 36 of file Release.cpp.

7.112.3.11 show()

```
std::string Release::show ( )  [virtual]
```

Reimplemented from ModelComponent.

Definition at line 24 of file Release.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Release.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Release.cpp

7.113 Remove Class Reference

```
#include <Remove.h>
```

Inheritance diagram for Remove:

Collaboration diagram for Remove:

Public Member Functions

- Remove (Model ∗model, std::string name="")
- virtual ∼Remove ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.113.1 Detailed Description

Remove module DESCRIPTION The Remove module removes a single entity from a specified position in a queue and sends it to a designated module. When an entity arrives at a Remove module, it removes the entity from the specified queue and sends it to the connected module. The rank of the entity signifies the location of the entity within the queue. The entity that caused the removal proceeds to the next module specified and is processed before the removed entity. TYPICAL USES Removing an order from a queue that is due to be completed next Calling a patient from a waiting room for an examination Retrieving the next order to be processed from a pile of documents Prompt Description Name Unique module identifier displayed on the module shape. Queue Name Name of the queue from which the entity will be removed. Rank of Entity Rank of the entity to remove from within the queue.

Definition at line 37 of file Remove.h.

7.113.2 Constructor & Destructor Documentation

7.113.2.1 Remove()

```
Remove::Remove (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Remove.cpp.

Here is the caller graph for this function:

7.113.2.2 ~Remove()

```
virtual Remove::~Remove ( ) [virtual], [default]
```

7.113.3 Member Function Documentation

7.113.3.1 _check()

```
bool Remove::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file Remove.cpp.

7.113.3.2 _execute()

```
void Remove::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file Remove.cpp.

Here is the call graph for this function:

7.113.3.3 _initBetweenReplications()

```
void Remove::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 47 of file Remove.cpp.

7.113.3.4 _loadInstance()

```
bool Remove::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 39 of file Remove.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.113.3.5 _saveInstance() `std::map< std::string, std::string > * Remove::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 50 of file Remove.cpp.

Here is the call graph for this function:

7.113.3.6 GetPluginInformation() `PluginInformation * Remove::GetPluginInformation ( ) [static]`

Definition at line 63 of file Remove.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.113.3.7 LoadInstance()

```
ModelComponent * Remove::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 24 of file Remove.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.113.3.8 show()

```
std::string Remove::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 20 of file Remove.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Remove.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Remove.cpp

7.114 Resource Class Reference

```
#include <Resource.h>
```

Inheritance diagram for Resource:

Collaboration diagram for Resource:

Public Types

- enum class ResourceState : int {
  IDLE = 1 , BUSY = 2 , FAILED = 3 , INACTIVE = 4 ,
  OTHER = 5 }
- typedef std::function< void(Resource *) > ResourceEventHandler
- typedef std::pair< std::pair< ResourceEventHandler, ModelComponent * >, unsigned int > SortedResourceEventHandler

Public Member Functions

- Resource (Model *model, std::string name="")
- virtual ~Resource ()=default
- virtual std::string show ()
- void seize (unsigned int quantity, double tnow)
- void release (unsigned int quantity, double tnow)
- void setResourceState (ResourceState _resourceState)
- Resource::ResourceState getResourceState () const
- void setCapacity (unsigned int _capacity)
- unsigned int getCapacity () const
- void setCostBusyHour (double _costBusyHour)

- double getCostBusyHour () const
- void setCostIdleHour (double _costIdleHour)
- double getCostIdleHour () const
- void setCostPerUse (double _costPerUse)
- double getCostPerUse () const
- unsigned int getNumberBusy () const
- void addReleaseResourceEventHandler (ResourceEventHandler eventHandler, ModelComponent *component, unsigned int priority)
- double getLastTimeSeized () const

Static Public Member Functions

- template<typename Class >
  static ResourceEventHandler SetResourceEventHandler (void(Class::*function)(Resource *), Class *object)
- static PluginInformation * GetPluginInformation ()
- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _createInternalElements ()
- virtual void _initBetweenReplications ()

Additional Inherited Members

7.114.1 Detailed Description

Resource module DESCRIPTION This data module defines the resources in the simulation system, including costing information and resource availability. Resources may have a fixed capacity that does not vary over the simulation run or may operate based on a schedule. Resource failures and states can also be specified in this module. TYPICAL USES Equipment (machinery, cash register, phone line) People (clerical, order processing, sales clerks, operators) PROMPTS Prompt Description Name The name of the resource whose characteristics are being defined. This name must be unique. Type Method for determining the capacity for a resource. Fixed Capacity will not change during the simulation run. Based on Schedule signifies that a Schedule module is used to specify the capacity and duration information for the resource. Capacity Number of resource units of a given name that are available to the system for processing. Applies only when Type is Fixed Capacity. Schedule Name Identifies the name of the schedule to be used by the resource. The schedule defines the capacity of a resource for a given period of time. Applies only when type is Schedule. Schedule Rule Dictates when the actual capacity change is to occur when a decrease in capacity is required for a busy resource unit. Applies only when Type is Schedule. Busy/Hour Cost per hour of a resource that is processing an entity. The resource becomes busy when it is originally allocated to an entity and becomes idle when it is released. During the time when it is busy, cost will accumulate based on the busy/hour cost. The busy cost per hour is automatically converted to the appropriate base time unit specified within the Replication Parameters page of the Run > Setup menu item. Idle/Hour Cost per hour of a resource that is idle. The resource is idle while it is not processing an entity. During the time when it is idle, cost will accumulate based on the idle/hour cost. The idle cost per hour is automatically converted to the appropriate base time unit specified within the Replication Parameters page of the Run > Setup menu item. Per Use Cost of a resource on a usage basis, regardless of the time for which it is used. Each time the resource is allocated to an entity, it will incur a per-use cost. StateSet Name Name of states that the resource may be assigned during the simulation run. Initial State Initial state of a resource. If specified, the name must be defined within the repeat group of state names. This field is shown only when a StateSet Name is defined. Failures Lists all failures that will be associated with the resource. Failure Name–Name of the failure associated with the resource. Failure Rule–Behavior that should occur when a failure is to occur for a busy resource unit. Report Statistics Specifies whether or not statistics will be collected automatically and stored in the report database for this resource.

Definition at line 81 of file Resource.h.

7.114.2 Member Typedef Documentation

7.114.2.1 ResourceEventHandler `typedef std::function<void(Resource*) > Resource::ResourceEventHandler`

Definition at line 83 of file Resource.h.

7.114.2.2 SortedResourceEventHandler `typedef std::pair<std::pair<ResourceEventHandler, ModelComponent*>, unsigned int> Resource::SortedResourceEventHandler`

Definition at line 84 of file Resource.h.

7.114.3 Member Enumeration Documentation

7.114.3.1 ResourceState `enum Resource::ResourceState : int [strong]`

Enumerator

IDLE	
BUSY	
FAILED	
INACTIVE	
OTHER	

Definition at line 90 of file Resource.h.

7.114.4 Constructor & Destructor Documentation

7.114.4.1 Resource()

```
Resource::Resource (
    Model * model,
    std::string name = "" )
```

Handlers are sorted by priority

Definition at line 18 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.4.2 ~Resource() `virtual Resource::~Resource ( ) [virtual], [default]`

7.114.5 Member Function Documentation

7.114.5.1 _check() `bool Resource::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 187 of file Resource.cpp.

7.114.5.2 _createInternalElements() `void Resource::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist before checking such expression

Reimplemented from ModelElement.

Definition at line 192 of file Resource.cpp.

Here is the call graph for this function:

7.114.5.3 _initBetweenReplications() `void Resource::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 71 of file Resource.cpp.

Here is the call graph for this function:

7.114.5.4 _loadInstance()

```
bool Resource::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 165 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.5.5 _saveInstance()

```
std::map< std::string, std::string > * Resource::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 177 of file Resource.cpp.

Here is the call graph for this function:

7.114.5.6 addReleaseResourceEventHandler()

```
void Resource::addReleaseResourceEventHandler (
    ResourceEventHandler eventHandler,
    ModelComponent * component,
    unsigned int priority )
```

Definition at line 124 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.5.7 getCapacity()

```
unsigned int Resource::getCapacity ( ) const
```

Definition at line 92 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.8 getCostBusyHour()

```
double Resource::getCostBusyHour ( ) const
```

Definition at line 100 of file Resource.cpp.

7.114.5.9 getCostIdleHour() `double Resource::getCostIdleHour ( ) const`

Definition at line 108 of file Resource.cpp.

7.114.5.10 getCostPerUse() `double Resource::getCostPerUse ( ) const`

Definition at line 116 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.11 getLastTimeSeized() `double Resource::getLastTimeSeized ( ) const`

Definition at line 139 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.12 getNumberBusy() `unsigned int Resource::getNumberBusy ( ) const`

Definition at line 120 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.13 GetPluginInformation() `PluginInformation * Resource::GetPluginInformation ( ) [static]`

Definition at line 150 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.5.14 getResourceState() `Resource::ResourceState Resource::getResourceState ( ) const`

Definition at line 84 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.15 LoadInstance()

```
ModelElement * Resource::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 155 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.5.16 release()

```
void Resource::release (
    unsigned int quantity,
    double tnow )
```

Definition at line 51 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.5.17 seize()

```
void Resource::seize (
    unsigned int quantity,
    double tnow )
```

Definition at line 42 of file Resource.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.114.5.18 setCapacity() `void Resource::setCapacity ( unsigned int _capacity )`

Definition at line 88 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.19 setCostBusyHour() `void Resource::setCostBusyHour ( double _costBusyHour )`

Definition at line 96 of file Resource.cpp.

7.114.5.20 setCostIdleHour() `void Resource::setCostIdleHour ( double _costIdleHour )`

Definition at line 104 of file Resource.cpp.

7.114.5.21 setCostPerUse()

```
void Resource::setCostPerUse (
    double _costPerUse )
```

Definition at line 112 of file Resource.cpp.

Here is the caller graph for this function:

7.114.5.22 SetResourceEventHandler()

```
template<typename Class >
static ResourceEventHandler Resource::SetResourceEventHandler (
    void(Class::*)(Resource *) function,
    Class * object )  [inline], [static]
```

Definition at line 86 of file Resource.h.

7.114.5.23 setResourceState()

```
void Resource::setResourceState (
    ResourceState _resourceState )
```

Definition at line 80 of file Resource.cpp.

7.114.5.24 show()

```
std::string Resource::show ( )  [virtual]
```

Reimplemented from ModelElement.

Definition at line 33 of file Resource.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Resource.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Resource.cpp

7.115 Sampler_if::RNG_Parameters Struct Reference

```
#include <Sampler_if.h>
```

Inheritance diagram for Sampler_if::RNG_Parameters:

Collaboration diagram for Sampler_if::RNG_Parameters:

Public Member Functions

- virtual ~RNG_Parameters ()=default

7.115.1 Detailed Description

class that encapsulates attributes required to generate random numbers, which depends on the generation method used.

Definition at line 26 of file Sampler_if.h.

7.115.2 Constructor & Destructor Documentation

7.115.2.1 ~RNG_Parameters() `virtual Sampler_if::RNG_Parameters::~RNG_Parameters ( ) [virtual], [default]`

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/Sampler_if.h

7.116 Route Class Reference

```
#include <Route.h>
```

Inheritance diagram for Route:

Collaboration diagram for Route:

Public Types

- enum class DestinationType : int { Station = 0 , BySequence = 1 }

Public Member Functions

- Route (Model ∗model, std::string name="")
- virtual ∼Route ()=default
- virtual std::string show ()
- void setStation (Station ∗_station)
- Station ∗ getStation () const
- void setRouteTimeExpression (std::string _routeTimeExpression)
- std::string getRouteTimeExpression () const
- void setRouteTimeTimeUnit (Util::TimeUnit _routeTimeTimeUnit)
- Util::TimeUnit getRouteTimeTimeUnit () const
- void setRouteDestinationType (DestinationType _routeDestinationType)
- Route::DestinationType getRouteDestinationType () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _createInternalElements ()

  This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Additional Inherited Members

7.116.1 Detailed Description

Route module DESCRIPTION The Route module transfers an entity to a specified station or the next station in the station visitation sequence defined for the entity. A delay time to transfer to the next station may be defined. When an entity enters the Route module, its Station attribute (Entity.Station) is set to the destination station. The entity is then sent to the destination station, using the route time specified. If the station destination is entered as By Sequence, the next station is determined by the entity's sequence and step within the set (defined by special-purpose attributes Entity.Sequence and Entity.Jobstep, respectively). TYPICAL USES Send a part to its next processing station based on its routing slip Send an account balance call to an account agent Send restaurant customers to a specific table PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Route Time Travel time from the entity's current location to the destination station. Units Time units for route-time parameters. Destination Type Method for determining the entity destination location. Selection of By Sequence requires that the entity has been assigned a sequence name and that the sequence itself has been defined. Station Name Name of the individual destination station. Attribute Name Name of the attribute that stores the station name to which entities will route. Expression Expression that is evaluated to the station name where entities will route.

Definition at line 53 of file Route.h.

7.116.2 Member Enumeration Documentation

7.116.2.1 DestinationType `enum Route::DestinationType : int [strong]`

Enumerator

Station	
BySequence	

Definition at line 56 of file Route.h.

7.116.3 Constructor & Destructor Documentation

7.116.3.1 Route()

```
Route::Route (
    Model * model,
    std::string name = "" )
```

Definition at line 21 of file Route.cpp.

Here is the caller graph for this function:

7.116.3.2 ~Route()

```
virtual Route::~Route ( ) [virtual], [default]
```

7.116.4 Member Function Documentation

7.116.4.1 _check()

```
bool Route::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 142 of file Route.cpp.

Here is the call graph for this function:

7.116.4.2 _createInternalElements() `void Route::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist when checking such expression.

Reimplemented from ModelComponent.

Definition at line 182 of file Route.cpp.

Here is the call graph for this function:

7.116.4.3 _execute() `void Route::_execute ( Entity * entity ) [protected], [virtual]`

Implements ModelComponent.

Definition at line 75 of file Route.cpp.

7.116.4.4 _initBetweenReplications() `void Route::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 127 of file Route.cpp.

Here is the call graph for this function:

7.116.4.5 _loadInstance()

```
bool Route::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 112 of file Route.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.116.4.6 _saveInstance()

```
std::map< std::string, std::string > * Route::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 131 of file Route.cpp.

Here is the call graph for this function:

7.116.4.7 GetPluginInformation() `PluginInformation * Route::GetPluginInformation ( ) [static]`

Definition at line 175 of file Route.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.116.4.8 getRouteDestinationType() `Route::DestinationType Route::getRouteDestinationType ( ) const`

Definition at line 71 of file Route.cpp.

7.116.4.9 getRouteTimeExpression() `std::string Route::getRouteTimeExpression ( ) const`

Definition at line 55 of file Route.cpp.

7.116.4.10 getRouteTimeTimeUnit() `Util::TimeUnit Route::getRouteTimeTimeUnit ( ) const`

Definition at line 63 of file Route.cpp.

7.116.4.11 getStation() `Station * Route::getStation ( ) const`

Definition at line 47 of file Route.cpp.

7.116.4.12 LoadInstance()

```
ModelComponent * Route::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 33 of file Route.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.116.4.13 setRouteDestinationType()

```
void Route::setRouteDestinationType (
    DestinationType _routeDestinationType )
```

Definition at line 67 of file Route.cpp.

Here is the caller graph for this function:

7.116.4.14 setRouteTimeExpression()

```
void Route::setRouteTimeExpression (
    std::string _routeTimeExpression )
```

Definition at line 51 of file Route.cpp.

Here is the caller graph for this function:

7.116.4.15 setRouteTimeTimeUnit()

```
void Route::setRouteTimeTimeUnit (
    Util::TimeUnit _routeTimeTimeUnit )
```

Definition at line 59 of file Route.cpp.

7.116.4.16 setStation()

```
void Route::setStation (
    Station * _station )
```

Definition at line 43 of file Route.cpp.

Here is the caller graph for this function:

7.116.4.17 show() `std::string Route::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 24 of file Route.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Route.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Route.cpp

7.117 Sampler_if Class Reference

```
#include <Sampler_if.h>
```

Inheritance diagram for Sampler_if:

Collaboration diagram for Sampler_if:

Classes

- struct RNG_Parameters

Public Member Functions

- virtual double random ()=0
- virtual double sampleBeta (double alpha, double beta, double infLimit, double supLimit)=0
- virtual double sampleBeta (double alpha, double beta)=0
- virtual double sampleErlang (double mean, int M)=0
- virtual double sampleExponential (double mean)=0
- virtual double sampleGamma (double mean, double alpha)=0
- virtual double sampleGumbell (double mode, double scale)=0
- virtual double sampleLogNormal (double mean, double stddev)=0
- virtual double sampleNormal (double mean, double stddev)=0
- virtual double sampleTriangular (double min, double mode, double max)=0
- virtual double sampleUniform (double min, double max)=0
- virtual double sampleWeibull (double alpha, double scale)=0
- virtual double sampleBinomial (int trials, double p)=0
- virtual double sampleBernoulli (double p)=0
- virtual double sampleDiscrete (double acumProb, double value,...)=0
- virtual double sampleDiscrete (double acumProb, double *prob, double *value, int size)=0
- virtual double sampleGeometric (double p)=0
- virtual void setRNGparameters (RNG_Parameters *param)=0
- virtual RNG_Parameters * getRNGparameters () const =0

7.117.1 Detailed Description

Interface that describes the methods to be implemented by classes that generate random values that follow a specific probability distribution.

Definition at line 20 of file Sampler_if.h.

7.117.2 Member Function Documentation

7.117.2.1 getRNGparameters() `virtual RNG_Parameters* Sampler_if::getRNGparameters ( ) const [pure virtual]`

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

7.117.2.2 random() `virtual double Sampler_if::random ( ) [pure virtual]`

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.3 sampleBernoulli() `virtual double Sampler_if::sampleBernoulli ( double p ) [pure virtual]`

Implemented in SamplerDefaultImpl1.

7.117.2.4 sampleBeta() [1/2] `virtual double Sampler_if::sampleBeta ( double alpha, double beta ) [pure virtual]`

Implemented in SamplerDefaultImpl1.

7.117.2.5 sampleBeta() [2/2]

```
virtual double Sampler_if::sampleBeta (
    double alpha,
    double beta,
    double infLimit,
    double supLimit ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.6 sampleBinomial()

```
virtual double Sampler_if::sampleBinomial (
    int trials,
    double p ) [pure virtual]
```

Implemented in SamplerDefaultImpl1.

7.117.2.7 sampleDiscrete() [1/2]

```
virtual double Sampler_if::sampleDiscrete (
    double acumProb,
    double * prob,
    double * value,
    int size ) [pure virtual]
```

Implemented in SamplerDefaultImpl1.

7.117.2.8 sampleDiscrete() [2/2]

```
virtual double Sampler_if::sampleDiscrete (
    double acumProb,
    double value,
    ... ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.9 sampleErlang()

```
virtual double Sampler_if::sampleErlang (
    double mean,
    int M ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.10 sampleExponential()

```
virtual double Sampler_if::sampleExponential (
    double mean ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.11 sampleGamma()

```
virtual double Sampler_if::sampleGamma (
    double mean,
    double alpha ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.12 sampleGeometric()

```
virtual double Sampler_if::sampleGeometric (
    double p ) [pure virtual]
```

Implemented in SamplerDefaultImpl1.

7.117.2.13 sampleGumbell()

```
virtual double Sampler_if::sampleGumbell (
    double mode,
    double scale ) [pure virtual]
```

Implemented in SamplerDefaultImpl1.

7.117.2.14 sampleLogNormal()

```
virtual double Sampler_if::sampleLogNormal (
    double mean,
    double stddev ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.15 sampleNormal()

```
virtual double Sampler_if::sampleNormal (
    double mean,
    double stddev ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.16 sampleTriangular()

```
virtual double Sampler_if::sampleTriangular (
    double min,
    double mode,
    double max ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.17 sampleUniform()

```
virtual double Sampler_if::sampleUniform (
    double min,
    double max ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.18 sampleWeibull()

```
virtual double Sampler_if::sampleWeibull (
    double alpha,
    double scale ) [pure virtual]
```

Implemented in SamplerDefaultImpl1, and SamplerBoostImpl.

Here is the caller graph for this function:

7.117.2.19 setRNGparameters()

```
virtual void Sampler_if::setRNGparameters (
    RNG_Parameters * param ) [pure virtual]
```

Implemented in SamplerBoostImpl, and SamplerDefaultImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/Sampler_if.h

7.118 SamplerBoostImpl Class Reference

```
#include <SamplerBoostImpl.h>
```

Inheritance diagram for SamplerBoostImpl:

Collaboration diagram for SamplerBoostImpl:

Classes

- struct BoostImplRNG_Parameters

Public Member Functions

- SamplerBoostImpl ()
- virtual ~SamplerBoostImpl ()=default
- virtual double random ()
- virtual double sampleBeta (double alpha, double beta, double infLimit, double supLimit)
- virtual double sampleDiscrete (double acumProb, double value,...)
- virtual double sampleErlang (double mean, int M)

- virtual double sampleExponential (double mean)
- virtual double sampleGamma (double mean, double alpha)
- virtual double sampleLogNormal (double mean, double stddev)
- virtual double sampleNormal (double mean, double stddev)
- virtual double sampleTriangular (double min, double mode, double max)
- virtual double sampleUniform (double min, double max)
- virtual double sampleWeibull (double alpha, double scale)
- void reset ()

  reinitialize seed and other parameters so (pseudo) random number sequence will be generated again.
- virtual void setRNGparameters (Sampler_if::RNG_Parameters *param)
- virtual RNG_Parameters * getRNGparameters () const

7.118.1 Detailed Description

Definition at line 20 of file SamplerBoostImpl.h.

7.118.2 Constructor & Destructor Documentation

7.118.2.1 SamplerBoostImpl() `SamplerBoostImpl::SamplerBoostImpl ( )`

Definition at line 18 of file SamplerBoostImpl.cpp.

7.118.2.2 ~SamplerBoostImpl() `virtual SamplerBoostImpl::~SamplerBoostImpl ( ) [virtual], [default]`

7.118.3 Member Function Documentation

7.118.3.1 getRNGparameters() `Sampler_if::RNG_Parameters * SamplerBoostImpl::getRNGparameters ( ) const [virtual]`

Implements Sampler_if.

Definition at line 71 of file SamplerBoostImpl.cpp.

7.118.3.2 random() `double SamplerBoostImpl::random ( ) [virtual]`

Implements Sampler_if.

Definition at line 21 of file SamplerBoostImpl.cpp.

7.118.3.3 reset() `void SamplerBoostImpl::reset ( )`

reinitialize seed and other parameters so (pseudo) random number sequence will be generated again.

7.118.3.4 sampleBeta()

```
double SamplerBoostImpl::sampleBeta (
    double alpha,
    double beta,
    double infLimit,
    double supLimit ) [virtual]
```

Implements Sampler_if.

Definition at line 26 of file SamplerBoostImpl.cpp.

7.118.3.5 sampleDiscrete()

```
double SamplerBoostImpl::sampleDiscrete (
    double acumProb,
    double value,
    ... ) [virtual]
```

Implements Sampler_if.

Definition at line 30 of file SamplerBoostImpl.cpp.

7.118.3.6 sampleErlang()

```
double SamplerBoostImpl::sampleErlang (
    double mean,
    int M ) [virtual]
```

Implements Sampler_if.

Definition at line 34 of file SamplerBoostImpl.cpp.

7.118.3.7 sampleExponential()

```
double SamplerBoostImpl::sampleExponential (
    double mean ) [virtual]
```

Implements Sampler_if.

Definition at line 38 of file SamplerBoostImpl.cpp.

7.118.3.8 sampleGamma()

```
double SamplerBoostImpl::sampleGamma (
    double mean,
    double alpha ) [virtual]
```

Implements Sampler_if.

Definition at line 42 of file SamplerBoostImpl.cpp.

7.118.3.9 sampleLogNormal()

```
double SamplerBoostImpl::sampleLogNormal (
    double mean,
    double stddev ) [virtual]
```

Implements Sampler_if.

Definition at line 46 of file SamplerBoostImpl.cpp.

7.118.3.10 sampleNormal()

```
double SamplerBoostImpl::sampleNormal (
    double mean,
    double stddev ) [virtual]
```

Implements Sampler_if.

Definition at line 50 of file SamplerBoostImpl.cpp.

7.118.3.11 sampleTriangular()

```
double SamplerBoostImpl::sampleTriangular (
    double min,
    double mode,
    double max ) [virtual]
```

Implements Sampler_if.

Definition at line 54 of file SamplerBoostImpl.cpp.

7.118.3.12 sampleUniform()

```
double SamplerBoostImpl::sampleUniform (
    double min,
    double max ) [virtual]
```

Implements Sampler_if.

Definition at line 58 of file SamplerBoostImpl.cpp.

7.118.3.13 sampleWeibull()

```
double SamplerBoostImpl::sampleWeibull (
    double alpha,
    double scale ) [virtual]
```

Implements Sampler_if.

Definition at line 63 of file SamplerBoostImpl.cpp.

7.118.3.14 setRNGparameters()

```
void SamplerBoostImpl::setRNGparameters (
    Sampler_if::RNG_Parameters * param ) [virtual]
```

Implements Sampler_if.

Definition at line 67 of file SamplerBoostImpl.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SamplerBoostImpl.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SamplerBoostImpl.cpp

7.119 SamplerDefaultImpl1 Class Reference

```
#include <SamplerDefaultImpl1.h>
```

Inheritance diagram for SamplerDefaultImpl1:

Collaboration diagram for SamplerDefaultImpl1:

Classes

- struct DefaultImpl1RNG_Parameters

Public Member Functions

- SamplerDefaultImpl1 ()
- virtual ~SamplerDefaultImpl1 ()=default
- virtual double random ()
- virtual double sampleBeta (double alpha, double beta, double infLimit, double supLimit)
- virtual double sampleBeta (double alpha, double beta)
- virtual double sampleErlang (double mean, int M)

- virtual double sampleExponential (double mean)
- virtual double sampleGamma (double mean, double alpha)
- virtual double sampleGumbell (double mode, double scale)
- virtual double sampleLogNormal (double mean, double stddev)
- virtual double sampleNormal (double mean, double stddev)
- virtual double sampleTriangular (double min, double mode, double max)
- virtual double sampleUniform (double min, double max)
- virtual double sampleWeibull (double alpha, double scale)
- virtual double sampleBinomial (int trials, double p)
- virtual double sampleBernoulli (double p)
- virtual double sampleDiscrete (double acumProb, double value,...)
- virtual double sampleDiscrete (double acumProb, double *prob, double *value, int size)
- virtual double sampleGeometric (double p)
- void reset ()

  reinitialize seed and other parameters so (pseudo) random number sequence will be generated again.
- virtual void setRNGparameters (RNG_Parameters *param)
- virtual RNG_Parameters * getRNGparameters () const

7.119.1 Detailed Description

Definition at line 20 of file SamplerDefaultImpl1.h.

7.119.2 Constructor & Destructor Documentation

7.119.2.1 SamplerDefaultImpl1() `SamplerDefaultImpl1::SamplerDefaultImpl1 ( )`

Definition at line 24 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.2.2 ~SamplerDefaultImpl1() `virtual SamplerDefaultImpl1::~SamplerDefaultImpl1 ( ) [virtual], [default]`

7.119.3 Member Function Documentation

7.119.3.1 getRNGparameters() `Sampler_if::RNG_Parameters * SamplerDefaultImpl1::getRNGparameters ( ) const [virtual]`

Implements Sampler_if.

Definition at line 38 of file SamplerDefaultImpl1.cpp.

7.119.3.2 random() `double SamplerDefaultImpl1::random ( ) [virtual]`

Implements Sampler_if.

Definition at line 42 of file SamplerDefaultImpl1.cpp.

Here is the caller graph for this function:

7.119.3.3 reset() `void SamplerDefaultImpl1::reset ( )`

reinitialize seed and other parameters so (pseudo) random number sequence will be generated again.

Definition at line 28 of file SamplerDefaultImpl1.cpp.

Here is the caller graph for this function:

7.119.3.4 sampleBernoulli()

```
double SamplerDefaultImpl1::sampleBernoulli (
    double p ) [virtual]
```

Implements Sampler_if.

Definition at line 230 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.5 sampleBeta() [1/2]

```
double SamplerDefaultImpl1::sampleBeta (
    double alpha,
    double beta ) [virtual]
```

Implements Sampler_if.

Definition at line 253 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.6 sampleBeta() [2/2]

```
double SamplerDefaultImpl1::sampleBeta (
    double alpha,
    double beta,
    double infLimit,
    double supLimit ) [virtual]
```

Implements Sampler_if.

Definition at line 206 of file SamplerDefaultImpl1.cpp.

7.119.3.7 sampleBinomial()

```
double SamplerDefaultImpl1::sampleBinomial (
    int trials,
    double p ) [virtual]
```

Implements Sampler_if.

Definition at line 216 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.8 sampleDiscrete() [1/2]

```
double SamplerDefaultImpl1::sampleDiscrete (
    double acumProb,
    double * prob,
    double * value,
    int size ) [virtual]
```

Implements Sampler_if.

Definition at line 109 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.9 sampleDiscrete() [2/2]

```
double SamplerDefaultImpl1::sampleDiscrete (
    double acumProb,
    double value,
    ... ) [virtual]
```

Implements Sampler_if.

Definition at line 104 of file SamplerDefaultImpl1.cpp.

7.119.3.10 sampleErlang()

```
double SamplerDefaultImpl1::sampleErlang (
    double mean,
    int M ) [virtual]
```

Implements Sampler_if.

Definition at line 59 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.11 sampleExponential()

```
double SamplerDefaultImpl1::sampleExponential (
    double mean ) [virtual]
```

Implements Sampler_if.

Definition at line 55 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.12 sampleGamma()

```
double SamplerDefaultImpl1::sampleGamma (
    double mean,
    double alpha ) [virtual]
```

Implements Sampler_if.

Definition at line 175 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.119.3.13 sampleGeometric()

```
double SamplerDefaultImpl1::sampleGeometric (
    double p ) [virtual]
```

Implements Sampler_if.

Definition at line 240 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.14 sampleGumbell()

```
double SamplerDefaultImpl1::sampleGumbell (
    double mode,
    double scale ) [virtual]
```

Implements Sampler_if.

Definition at line 247 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.15 sampleLogNormal()

```
double SamplerDefaultImpl1::sampleLogNormal (
    double mean,
    double stddev ) [virtual]
```

Implements Sampler_if.

Definition at line 83 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.16 sampleNormal()

```
double SamplerDefaultImpl1::sampleNormal (
    double mean,
    double stddev ) [virtual]
```

Implements Sampler_if.

Definition at line 70 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.119.3.17 sampleTriangular()

```
double SamplerDefaultImpl1::sampleTriangular (
    double min,
    double mode,
    double max ) [virtual]
```

Implements Sampler_if.

Definition at line 91 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.18 sampleUniform()

```
double SamplerDefaultImpl1::sampleUniform (
    double min,
    double max ) [virtual]
```

Implements Sampler_if.

Definition at line 51 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.19 sampleWeibull()

```
double SamplerDefaultImpl1::sampleWeibull (
    double alpha,
    double scale ) [virtual]
```

Implements Sampler_if.

Definition at line 211 of file SamplerDefaultImpl1.cpp.

Here is the call graph for this function:

7.119.3.20 setRNGparameters()

```
void SamplerDefaultImpl1::setRNGparameters (
    Sampler_if::RNG_Parameters * param ) [virtual]
```

Implements Sampler_if.

Definition at line 33 of file SamplerDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.cpp

7.120 ScenarioExperiment_if Class Reference

```
#include <ScenarioExperiment_if.h>
```

Collaboration diagram for ScenarioExperiment_if:

7.120.1 Detailed Description

Definition at line 17 of file ScenarioExperiment_if.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/ScenarioExperiment_if.h

7.121 Schedule Class Reference

```
#include <Schedule.h>
```

Inheritance diagram for Schedule:

Collaboration diagram for Schedule:

Public Member Functions

- Schedule (Model ∗model, std::string name="")
- virtual ∼Schedule ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.121.1 Detailed Description

Schedule module DESCRIPTION This data module may be used in conjunction with the Resource module to define an operating schedule for a resource or with the Create module to define an arrival schedule. Additionally, a schedule may be used and referenced to factor time delays based on the simulation time. TYPICAL USES Work schedule for staff, including breaks Breakdown patterns for equipment Volume of customers arriving at a store Learning-curve factors for new workers PROMPTS File Read Time Specifies when to read the values from the file into the variable. If you select PreCheck, the values for the variable are read while the model is still in Edit mode (prior to the model being checked and compiled). If you select BeginSimulation, values are read when the model is compiled, prior to the first replication. If you select BeginReplication, values are read prior to each replication. Initial Values Lists the initial value or values of the variable. You can assign new values to the variable at different stages of the model by using the Assign module. Initial Value Variable value at the start of the simulation. Prompt Description Name The name of the schedule being defined. This name must be unique. Type Type of schedule being defined. This may be Capacity-related (for resource schedules), Arrival-related (for the Create module), or Other (miscellaneous time delays or factors) Time Units Time units used for the time-duration information. Scale Factor Method of scaling the schedule for increases or decreases in Arrival/Other values. The specified Value fields will be multiplied by the scale factor to determine the new values. Not available for Capacity-type schedules. Durations Lists the value and duration pairs for the schedule. Values can be capacity, arrival, or other type values, while the duration is specified in time units. Schedule pairs will repeat after all durations have been completed, unless the last duration is left blank (infinite). Schedule data can be entered graphically using the graphical schedule editor or manually using the Value/ Duration fields. Value Represents either the capacity of a resource (if Type is Capacity), arrival rate (if Type is Arrival), or some other value (if Type is Other). Examples of Other may be a factor that is used in a delay expression to scale a delay time during various parts of the day. Duration Time duration for which a specified Value will be valid.

Definition at line 72 of file Schedule.h.

7.121.2 Constructor & Destructor Documentation

7.121.2.1 Schedule()

```
Schedule::Schedule (
    Model * model,
    std::string name = "" )
```

Definition at line 16 of file Schedule.cpp.

7.121.2.2 ~Schedule()

```
virtual Schedule::~Schedule ( ) [virtual], [default]
```

7.121.3 Member Function Documentation

7.121.3.1 _check()

```
bool Schedule::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 34 of file Schedule.cpp.

7.121.3.2 _loadInstance()

```
bool Schedule::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 28 of file Schedule.cpp.

7.121.3.3 _saveInstance()

```
std::map< std::string, std::string > * Schedule::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 31 of file Schedule.cpp.

7.121.3.4 GetPluginInformation()

```
PluginInformation * Schedule::GetPluginInformation ( ) [static]
```

Definition at line 22 of file Schedule.cpp.

Here is the caller graph for this function:

7.121.3.5 LoadInstance()

```
ModelElement * Schedule::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file Schedule.cpp.

7.121.3.6 show() `std::string Schedule::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 19 of file Schedule.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Schedule.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Schedule.cpp

7.122 Search Class Reference

```
#include <Search.h>
```

Inheritance diagram for Search:

Collaboration diagram for Search:

Public Member Functions

- Search (Model ∗model, std::string name="")
- virtual ∼Search ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.122.1 Detailed Description

Search module DESCRIPTION The Search module searches a queue, a group (batch), or an expression to find the entity rank (for entities in a queue or group) or the value of the global variable J that satisfies the specified search condition. When searching a queue or group, the value of the global system variable J is set to the rank of the first entity that satisfies Search Condition, or to 0 if Search Condition is not satisfied. When searching an expression, the global system variable J is set to the value of the first index value that satisfies the search condition or to zero if no value of J in the specified range satisfies the search condition. When an entity arrives at a Search module, the index J is set to the starting index and the search condition is then checked. If the search condition is satisfied, the search ends and the current value of J is retained. Otherwise, the value of J is increased or decreased and the condition is rechecked. This process repeats until the search condition is satisfied or the ending value is reached. If the condition is not met or there are no entities in the queue or group, J is set equal to 0. TYPICAL USES Looking for a particular order number in a queue Searching a group for a certain part type Determining which process to enter based on availability of resources (search an expression) Prompt Description Name Unique module identifier displayed on the module shape. Type Determination of what will be searched. Search options include entities in a queue, entities within a group (batch) or some expression(s). Queue Name Name of the queue that will be searched. Applies only when the Type is Search a Queue. Starting Value Starting rank in the queue or group or starting value for J in an expression. Ending Value Ending rank in the queue or group or ending value for J in an expression. Search Condition Condition containing the index J for searching expressions or containing an attribute name(s) for searching queues or batches.

Definition at line 55 of file Search.h.

7.122.2 Constructor & Destructor Documentation

7.122.2.1 Search()

```
Search::Search (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Search.cpp.

Here is the caller graph for this function:

7.122.2.2 ~Search()

```
virtual Search::~Search ( ) [virtual], [default]
```

7.122.3 Member Function Documentation

7.122.3.1 _check()

```
bool Search::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file Search.cpp.

7.122.3.2 _execute()

```
void Search::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file Search.cpp.

Here is the call graph for this function:

7.122.3.3 _initBetweenReplications()

```
void Search::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 47 of file Search.cpp.

7.122.3.4 _loadInstance()

```
bool Search::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 39 of file Search.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.122.3.5 _saveInstance()

```
std::map< std::string, std::string > * Search::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 50 of file Search.cpp.

Here is the call graph for this function:

7.122.3.6 GetPluginInformation() `PluginInformation * Search::GetPluginInformation ( ) [static]`

Definition at line 63 of file Search.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.122.3.7 LoadInstance() `ModelComponent * Search::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 24 of file Search.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.122.3.8 show() `std::string Search::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 20 of file Search.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Search.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Search.cpp

7.123 SeizableItem Class Reference

```
#include <SeizableItem.h>
```

Collaboration diagram for SeizableItem:

Public Types

- enum class SelectionRule : int {
  CYCLICAL = 1 , RANDOM = 2 , SPECIFICMEMBER = 3 , LARGESTREMAININGCAPACITY = 4 , SMALLESTNUMBERBUSY = 5 }
- enum class SeizableType : int { RESOURCE = 1 , SET = 2 }

Public Member Functions

- SeizableItem (ModelElement *resourceOrSet, std::string quantityExpression="1", SeizableItem::SelectionRule selectionRule=SeizableItem::SelectionRule::LARGESTREMAININGCAPACITY, std::string saveAttribute="", std::string index="0")
- bool loadInstance (std::map< std::string, std::string > *fields, unsigned int parentIndex)
- std::map< std::string, std::string > * saveInstance (unsigned int parentIndex)
- bool loadInstance (std::map< std::string, std::string > *fields)
- std::map< std::string, std::string > * saveInstance ()
- std::string show ()
- void setIndex (std::string index)
- std::string getIndex () const
- void setSaveAttribute (std::string saveAttribute)
- std::string getSaveAttribute () const
- void setSelectionRule (SelectionRule selectionRule)
- SelectionRule getSelectionRule () const
- void setQuantityExpression (std::string quantityExpression)
- std::string getQuantityExpression () const
- std::string getResourceName () const
- void setResource (Resource *resource)
- Resource * getResource () const
- void setSet (Set *set)
- Set * getSet () const
- void setSeizableType (SeizableType resourceType)
- SeizableType getSeizableType () const
- void setLastMemberSeized (unsigned int lastMemberSeized)
- unsigned int getLastMemberSeized () const
- ModelElement * getSeizable () const
- void setElementManager (ElementManager *_elementManager)

7.123.1 Detailed Description

Definition at line 23 of file SeizableItem.h.

7.123.2 Member Enumeration Documentation

7.123.2.1 SeizableType `enum SeizableItem::SeizableType : int [strong]`

Enumerator

RESOURCE	
SET	

Definition at line 30 of file SeizableItem.h.

7.123.2.2 SelectionRule `enum SeizableItem::SelectionRule : int [strong]`

Enumerator

CYCLICAL	
RANDOM	
SPECIFICMEMBER	
LARGESTREMAININGCAPACITY	
SMALLESTNUMBERBUSY	

Definition at line 26 of file SeizableItem.h.

7.123.3 Constructor & Destructor Documentation

7.123.3.1 SeizableItem()

```
SeizableItem::SeizableItem (
    ModelElement * resourceOrSet,
    std::string quantityExpression = "1",
    SeizableItem::SelectionRule selectionRule = SeizableItem::SelectionRule::LARGESTREMAININGCAPACITY
    std::string saveAttribute = "",
    std::string index = "0" )
```

Definition at line 19 of file SeizableItem.cpp.

7.123.4 Member Function Documentation

7.123.4.1 getIndex() `std::string SeizableItem::getIndex ( ) const`

Definition at line 128 of file SeizableItem.cpp.

7.123.4.2 getLastMemberSeized() `unsigned int SeizableItem::getLastMemberSeized ( ) const`

Definition at line 190 of file SeizableItem.cpp.

7.123.4.3 getQuantityExpression() `std::string SeizableItem::getQuantityExpression ( ) const`

Definition at line 152 of file SeizableItem.cpp.

Here is the caller graph for this function:

7.123.4.4 getResource() `Resource * SeizableItem::getResource ( ) const`

Definition at line 165 of file SeizableItem.cpp.

Here is the caller graph for this function:

7.123.4.5 getResourceName() `std::string SeizableItem::getResourceName ( ) const`

Definition at line 156 of file SeizableItem.cpp.

7.123.4.6 getSaveAttribute() `std::string SeizableItem::getSaveAttribute ( ) const`

Definition at line 136 of file SeizableItem.cpp.

7.123.4.7 getSeizable() `ModelElement * SeizableItem::getSeizable ( ) const`

Definition at line 194 of file SeizableItem.cpp.

7.123.4.8 getSeizableType() `SeizableItem::SeizableType SeizableItem::getSeizableType ( ) const`

Definition at line 182 of file SeizableItem.cpp.

Here is the caller graph for this function:

7.123.4.9 getSelectionRule() `SeizableItem::SelectionRule SeizableItem::getSelectionRule ( ) const`

Definition at line 144 of file SeizableItem.cpp.

7.123.4.10 getSet() `Set * SeizableItem::getSet ( ) const`

Definition at line 174 of file SeizableItem.cpp.

Here is the caller graph for this function:

7.123.4.11 loadInstance() [1/2] `bool SeizableItem::loadInstance ( std::map< std::string, std::string > * fields )`

Definition at line 48 of file SeizableItem.cpp.

Here is the call graph for this function:

7.123.4.12 loadInstance() [2/2]

```
bool SeizableItem::loadInstance (
    std::map< std::string, std::string > * fields,
    unsigned int parentIndex )
```

Definition at line 71 of file SeizableItem.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.123.4.13 saveInstance() [1/2]

```
std::map< std::string, std::string > * SeizableItem::saveInstance ( )
```

Definition at line 108 of file SeizableItem.cpp.

Here is the call graph for this function:

7.123.4.14 saveInstance() [2/2]

```
std::map< std::string, std::string > * SeizableItem::saveInstance (
            unsigned int parentIndex )
```

Definition at line 95 of file SeizableItem.cpp.

Here is the call graph for this function:

7.123.4.15 setElementManager()

```
void SeizableItem::setElementManager (
            ElementManager * _elementManager )
```

Definition at line 198 of file SeizableItem.cpp.

Here is the caller graph for this function:

7.123.4.16 setIndex()

```
void SeizableItem::setIndex (
            std::string index )
```

Definition at line 124 of file SeizableItem.cpp.

7.123.4.17 setLastMemberSeized()

```
void SeizableItem::setLastMemberSeized (
            unsigned int lastMemberSeized )
```

Definition at line 186 of file SeizableItem.cpp.

7.123.4.18 setQuantityExpression()

```
void SeizableItem::setQuantityExpression (
    std::string quantityExpression )
```

Definition at line 148 of file SeizableItem.cpp.

7.123.4.19 setResource()

```
void SeizableItem::setResource (
    Resource * resource )
```

Definition at line 160 of file SeizableItem.cpp.

Here is the call graph for this function:

7.123.4.20 setSaveAttribute()

```
void SeizableItem::setSaveAttribute (
    std::string saveAttribute )
```

Definition at line 132 of file SeizableItem.cpp.

7.123.4.21 setSeizableType()

```
void SeizableItem::setSeizableType (
    SeizableItem::SeizableType resourceType )
```

Definition at line 178 of file SeizableItem.cpp.

7.123.4.22 setSelectionRule()

```
void SeizableItem::setSelectionRule (
    SeizableItem::SelectionRule selectionRule )
```

Definition at line 140 of file SeizableItem.cpp.

7.123.4.23 setSet() `void SeizableItem::setSet ( Set * set )`

Definition at line 169 of file SeizableItem.cpp.

Here is the call graph for this function:

7.123.4.24 show() `std::string SeizableItem::show ( )`

Definition at line 120 of file SeizableItem.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/SeizableItem.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/SeizableItem.cpp

7.124 Seize Class Reference

```
#include <Seize.h>
```

Inheritance diagram for Seize:

Collaboration diagram for Seize:

Public Member Functions

- Seize (Model *model, std::string name="")
- virtual ~Seize ()=default
- virtual std::string show ()
- void setPriority (unsigned short _priority)
- unsigned short getPriority () const
- void setAllocationType (unsigned int _allocationType)
- unsigned int getAllocationType () const
- void setQueue (Queue *queue)

  Deprected.
- List< SeizableItem * > * getSeizeRequests () const
- void setQueueableItem (QueueableItem *_queueableItem)
- QueueableItem * getQueueableItem () const
- void setSaveAttribute (std::string _saveAttribute)
- std::string getSaveAttribute () const

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelComponent * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)

Additional Inherited Members

7.124.1 Detailed Description

Seize module DESCRIPTION The Seize module allocates units of one or more resources to an entity. The Seize module may be used to seize units of a particular resource, a member of a resource set, or a resource as defined by an alternative method, such as an attribute or expression. When an entity enters this module, it waits in a queue (if specified) until all specified resources are available simultaneously. Allocation type for resource usage is also specified. TYPICAL USES Beginning a customer order (seize the operator) Starting a tax return (seize the accountant) Being admitted to hospital (seize the hospital room, nurse, doctor) PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Allocation Determines to which category the resource usage cost will be allocated for an entity going through the Seize module. Priority Priority value of the entity waiting at this module for the resource(s) specified if one or more entities from other modules are waiting for the same resource(s). Type Type of resource for seizing, either specifying a particular resource, or selecting from a pool of resources (that is, a resource set). The name of the resource may also be specified as an attribute value or within an expression. Resource Name Name of the resource that will be seized. Set Name Name of the resource set from which a member will be seized. Attribute Name Name of the attribute that stores the resource name to be seized. Expression Expression that evaluates to a resource name to be seized. Quantity Number of resources of a given name or from a given set that will be seized. For sets, this value specifies only the number of a selected resource that will be seized (based on the resource's capacity), not the number of members to be seized within the set. Selection Rule Method of selecting among available resources in a set. Cyclical will cycle through available members (for example, 1-2-3-1-2- 3). Random will randomly select a member. Preferred Order will always select the first available member (for example, 1, if available; then 2, if available; then 3). Specific Member requires an input attribute value to specify which member of the set (previously saved in the Save Attribute field). Largest Remaining Capacity and Smallest Number Busy are used for resources with multiple capacity. Save Attribute Attribute name used to store the index number into the set of the member that is chosen. This attribute can later be referenced with the Specific Member selection rule. Set Index Index value into the set that identifies the number into the set of the member requested. If an attribute name is used, the entity must have a value for the attribute before utilizing this option. Resource State State of the resource that will be assigned after the resource is seized. The resource state must be defined with the Resource module. Queue Type Determines the type of queue used to hold the entities while waiting to seize the resource(s). If Queue is selected, the queue name is specified. If Set is selected, the queue set and member in the set are specified. If Internal is selected, an internal queue is used to hold all waiting entities. Attribute and Expression are additional methods for defining the queue to be used. Queue Name This field is visible only if Queue Type is Queue, and it defines the symbol name of the queue. Set Name This field is visible only if Queue Type is Set, and it defines the queue set that contains the queue being referenced. Set Index This field is visible only if Queue Type is Set, and it defines the index into the queue set. Note that this is the index into the set and not the name of the queue in the set. For example, the only valid entries for a queue set containing three members is an expression that evaluates to 1, 2, or 3. Attribute This field is visible only if Queue Type is Attribute. The attribute entered in this field will be evaluated to indicate which queue is to be used. Expression This field is visible only if Queue Type is Expression. The expression entered in this field will be evaluated to indicate which queue is to be used.

Definition at line 122 of file Seize.h.

7.124.2 Constructor & Destructor Documentation

7.124.2.1 Seize()

```
Seize::Seize (
    Model * model,
    std::string name = "" )
```

Definition at line 20 of file Seize.cpp.

Here is the caller graph for this function:

7.124.2.2 ~Seize()

```
virtual Seize::~Seize ( ) [virtual], [default]
```

7.124.3 Member Function Documentation

7.124.3.1 _check()

```
bool Seize::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 281 of file Seize.cpp.

Here is the call graph for this function:

7.124.3.2 _execute()

```
void Seize::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 203 of file Seize.cpp.

7.124.3.3 _initBetweenReplications()

```
void Seize::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 229 of file Seize.cpp.

7.124.3.4 _loadInstance()

```
bool Seize::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 237 of file Seize.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.124.3.5 _saveInstance() `std::map< std::string, std::string > * Seize::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 262 of file Seize.cpp.

Here is the call graph for this function:

7.124.3.6 getAllocationType() `unsigned int Seize::getAllocationType ( ) const`

Definition at line 47 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.7 GetPluginInformation() `PluginInformation * Seize::GetPluginInformation ( ) [static]`

Definition at line 313 of file Seize.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.124.3.8 getPriority() `unsigned short Seize::getPriority ( ) const`

Definition at line 39 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.9 getQueueableItem() `QueueableItem * Seize::getQueueableItem ( ) const`

Definition at line 156 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.10 getSaveAttribute() `std::string Seize::getSaveAttribute ( ) const`

Definition at line 164 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.11 getSeizeRequests() `List< SeizableItem * > * Seize::getSeizeRequests ( ) const`

Definition at line 148 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.12 LoadInstance() `ModelComponent * Seize::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 320 of file Seize.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.124.3.13 setAllocationType() `void Seize::setAllocationType ( unsigned int _allocationType )`

Definition at line 43 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.14 setPriority() `void Seize::setPriority ( unsigned short _priority )`

Definition at line 35 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.15 setQueue() `void Seize::setQueue (Queue * queue )`

Deprected.

Definition at line 97 of file Seize.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.124.3.16 setQueueableItem() `void Seize::setQueueableItem (QueueableItem * _queueableItem )`

Definition at line 152 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.17 setSaveAttribute() `void Seize::setSaveAttribute ( std::string _saveAttribute )`

Definition at line 160 of file Seize.cpp.

Here is the caller graph for this function:

7.124.3.18 show() `std::string Seize::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 23 of file Seize.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Seize.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Seize.cpp

7.125 yy::genesyspp_parser::semantic_type Class Reference

```
#include <GenesysParser.h>
```

Collaboration diagram for yy::genesyspp_parser::semantic_type:

Public Types

- typedef semantic_type self_type

  Type of ∗this.

Public Member Functions

- semantic_type () YY_NOEXCEPT

  Empty construction.
- template<typename T >
  semantic_type (YY_RVREF(T) t)

  Construct and fill.
- ~semantic_type () YY_NOEXCEPT

  Destruction, allowed only if empty.
- template<typename T >
  T & emplace ()

  Instantiate an empty T in here.
- template<typename T >
  T & emplace (const T &t)

  Instantiate a T in here from t.
- template<typename T >
  T & build ()
- template<typename T >
  T & build (const T &t)
- template<typename T >
  T & as () YY_NOEXCEPT

  Accessor to a built T.
- template<typename T >
  const T & as () const YY_NOEXCEPT

  Const accessor to a built T (for printer).
- template<typename T >
  void swap (self_type &that) YY_NOEXCEPT
- template<typename T >
  void move (self_type &that)
- template<typename T >
  void copy (const self_type &that)

  Copy the content of that to this.
- template<typename T >
  void destroy ()

  Destroy the stored T.

7.125.1 Detailed Description

A buffer to store and retrieve objects.

Sort of a variant, but does not keep track of the nature of the stored data, since that knowledge is available via the current parser state.

Definition at line 231 of file GenesysParser.h.

7.125.2 Member Typedef Documentation

7.125.2.1 self_type `typedef semantic_type yy::genesyspp_parser::semantic_type::self_type`

Type of ∗this.

Definition at line 235 of file GenesysParser.h.

7.125.3 Constructor & Destructor Documentation

7.125.3.1 semantic_type() [1/2] `yy::genesyspp_parser::semantic_type::semantic_type ( ) [inline]`

Empty construction.

Definition at line 238 of file GenesysParser.h.

7.125.3.2 semantic_type() [2/2] `template<typename T >`
`yy::genesyspp_parser::semantic_type::semantic_type (`
`YY_RVREF(T) t ) [inline]`

Construct and fill.

Definition at line 245 of file GenesysParser.h.

7.125.3.3 ∼semantic_type() `yy::genesyspp_parser::semantic_type::∼semantic_type ( ) [inline]`

Destruction, allowed only if empty.

Definition at line 260 of file GenesysParser.h.

7.125.4 Member Function Documentation

7.125.4.1 as() [1/2] `template<typename T >`
`const T& yy::genesyspp_parser::semantic_type::as ( ) const [inline]`

Const accessor to a built *T* (for printer).

Definition at line 332 of file GenesysParser.h.

7.125.4.2 as() [2/2] template<typename T >

```
T& yy::genesyspp_parser::semantic_type::as ( ) [inline]
```

Accessor to a built *T*.

Definition at line 321 of file GenesysParser.h.

Here is the caller graph for this function:

7.125.4.3 build() [1/2] template<typename T >

```
T& yy::genesyspp_parser::semantic_type::build ( ) [inline]
```

Instantiate an empty *T* in here. Obsolete, use emplace.

Definition at line 304 of file GenesysParser.h.

7.125.4.4 build() [2/2] template<typename T >

```
T& yy::genesyspp_parser::semantic_type::build (
            const T & t ) [inline]
```

Instantiate a *T* in here from *t*. Obsolete, use emplace.

Definition at line 313 of file GenesysParser.h.

7.125.4.5 copy() template<typename T >

```
void yy::genesyspp_parser::semantic_type::copy (
            const self_type & that ) [inline]
```

Copy the content of *that* to this.

Definition at line 387 of file GenesysParser.h.

Here is the call graph for this function:

Here is the caller graph for this function:

7.125.4.6 destroy() `template<typename T >`

`void yy::genesyspp_parser::semantic_type::destroy ( ) [inline]`

Destroy the stored *T*.

Definition at line 395 of file GenesysParser.h.

Here is the caller graph for this function:

7.125.4.7 emplace() [1/2] `template<typename T >`

`T& yy::genesyspp_parser::semantic_type::emplace ( ) [inline]`

Instantiate an empty *T* in here.

Definition at line 280 of file GenesysParser.h.

7.125.4.8 emplace() [2/2] `template<typename T >`

```
T& yy::genesyspp_parser::semantic_type::emplace (
            const T & t ) [inline]
```

Instantiate a *T* in here from *t*.

Definition at line 291 of file GenesysParser.h.

7.125.4.9 move() template<typename T >

```
void yy::genesyspp_parser::semantic_type::move (
            self_type & that )  [inline]
```

Move the content of *that* to this.

Destroys *that*.

Definition at line 362 of file GenesysParser.h.

Here is the call graph for this function:

7.125.4.10 swap() template<typename T >

```
void yy::genesyspp_parser::semantic_type::swap (
            self_type & that )  [inline]
```

Swap the content with *that*, of same type.

Both variants must be built beforehand, because swapping the actual data requires reading it (with as()), and this is not possible on unconstructed variants: it would require some dynamic testing, which should not be the variant's responsibility. Swapping between built and (possibly) non-built is done with self_type::move ().

Definition at line 350 of file GenesysParser.h.

7.125.5 Member Data Documentation

7.125.5.1 yyalign_me long double yy::genesyspp_parser::semantic_type::yyalign_me

Strongest alignment constraints.

Definition at line 532 of file GenesysParser.h.

7.125.5.2 yyraw `char yy::genesyspp_parser::semantic_type::yyraw[size]`

A buffer large enough to store any of the semantic values.

Definition at line 534 of file GenesysParser.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.126 Separate Class Reference

```
#include <Separate.h>
```

Inheritance diagram for Separate:

Collaboration diagram for Separate:

Public Member Functions

- Separate (Model ∗model, std::string name="")
- virtual ∼Separate ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.126.1 Detailed Description

Separate module DESCRIPTION This module can be used to either copy an incoming entity into multiple entities or to split a previously batched entity. Rules for allocating costs and times to the duplicate are also specified. Rules for attribute assignment to member entities are specified as well. When splitting existing batches, the temporary representative entity that was formed is disposed and the original entities that formed the group are recovered. The entities proceed sequentially from the module in the same order in which they originally were added to the batch. When duplicating entities, the specified number of copies is made and sent from the module. The original incoming entity also leaves the module. TYPICAL USES Send individual entities to represent boxes removed from a container Send an order both to fulfillment and billing for parallel processing Separate a previously batched set of documents PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Method of separating the incoming entity. Duplicate Original will simply take the original entity and make some number of identical duplicates. Split Existing Batch requires that the incoming entity be a temporarily batched entity using the Batch module. The original entities from the batch will be split. Percent Cost to Duplicates Allocation of costs and times of the incoming entity to the outgoing duplicates. This value is specified as a percentage of the original entity's costs and times (between 0-100). The percentage specified will be split evenly between the duplicates, while the original entity will retain any remaining cost/time percentage. Visible only when Type is Duplicate Original.

of Duplicates Number of outgoing entities that will leave the module, in

addition to the original incoming entity. Applies only when Type is Duplicate Original. Member Attributes Method of determining how to assign the representative entity attribute values to the original entities.These options relate to six of the special-purpose attributes (Entity.Type, Entity.Picture, Entity.Sequence, Entity.Station, Entity.Jobstep, and Entity.HoldCostRate) and all user-defined attributes. Applies only when Type is Split Existing Batch. Attribute Name Name of representative entity attribute(s) that are assigned to original entities of the group. Applies only when Member Attributes is Take Specific Representative Values.

Definition at line 65 of file Separate.h.

7.126.2 Constructor & Destructor Documentation

7.126.2.1 Separate()

```
Separate::Separate (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Separate.cpp.

Here is the caller graph for this function:

7.126.2.2 ~Separate() `virtual Separate::~Separate ( ) [virtual], [default]`

7.126.3 Member Function Documentation

7.126.3.1 _check() `bool Separate::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 58 of file Separate.cpp.

7.126.3.2 _execute() `void Separate::_execute ( Entity * entity ) [protected], [virtual]`

Implements ModelComponent.

Definition at line 35 of file Separate.cpp.

Here is the call graph for this function:

7.126.3.3 _initBetweenReplications() `void Separate::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 49 of file Separate.cpp.

7.126.3.4 _loadInstance()

```
bool Separate::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 41 of file Separate.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.126.3.5 _saveInstance()

```
std::map< std::string, std::string > * Separate::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 52 of file Separate.cpp.

Here is the call graph for this function:

7.126.3.6 GetPluginInformation() `PluginInformation * Separate::GetPluginInformation ( ) [static]`

Definition at line 65 of file Separate.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.126.3.7 LoadInstance() `ModelComponent * Separate::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 25 of file Separate.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.126.3.8 show() `std::string Separate::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file Separate.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Separate.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Separate.cpp

7.127 Sequence Class Reference

```
#include <Sequence.h>
```

Inheritance diagram for Sequence:

Collaboration diagram for Sequence:

Public Member Functions

- Sequence (Model ∗model, std::string name="")
- virtual ∼Sequence ()=default
- virtual std::string show ()
- List< SequenceStep ∗ > ∗ getSteps () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.127.1 Detailed Description

Sequence module DESCRIPTION The Sequence module is used to define a sequence for entity flow through the model. A sequence consists of an ordered list of stations that an entity will visit. For each station in the visitation sequence, attributes and variables may be assigned values. Each station in the visitation sequence is referred to as a step (or jobstep) in the sequence. Three special-purpose attributes are provided for all entities. The Sequence attribute (Entity.Sequence) defines the sequence that an entity is to follow; a value of 0 indicates that the entity is not following any sequence. In order for an entity to follow a sequence, its Sequence attribute must be assigned a value (for example, in the Assign module). The Jobstep attribute (Entity.Jobstep) stores the entity's current step number in the sequence. This value is updated automatically each time an entity is transferred. You typically do not need to assign explicitly a value to Jobstep in the model. The PlannedStation attribute (Entity.PlannedStation) stores the number of the station associated with the next jobstep in the sequence. This attribute is not user-assignable. It is automatically updated whenever Entity.Sequence or Entity.JobStep changes, or whenever the entity enters a station. Jobstep names must be globally unique. TYPICAL USES Define a routing path for part processing Define a sequence of steps patients must take upon arrival at an emergency room

Definition at line 100 of file Sequence.h.

7.127.2 Constructor & Destructor Documentation

7.127.2.1 Sequence()

```
Sequence::Sequence (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Sequence.cpp.

Here is the caller graph for this function:

7.127.2.2 ~Sequence()

```
virtual Sequence::~Sequence ( ) [virtual], [default]
```

7.127.3 Member Function Documentation

7.127.3.1 _check()

```
bool Sequence::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 60 of file Sequence.cpp.

Here is the call graph for this function:

7.127.3.2 _loadInstance()

```
bool Sequence::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 45 of file Sequence.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.127.3.3 _saveInstance() `std::map< std::string, std::string > * Sequence::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 55 of file Sequence.cpp.

Here is the call graph for this function:

7.127.3.4 GetPluginInformation() `PluginInformation * Sequence::GetPluginInformation ( ) [static]`

Definition at line 26 of file Sequence.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.127.3.5 getSteps() `List< SequenceStep * > * Sequence::getSteps ( ) const`

Definition at line 41 of file Sequence.cpp.

Here is the caller graph for this function:

7.127.3.6 LoadInstance()

```
ModelElement * Sequence::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 31 of file Sequence.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.127.3.7 show()

```
std::string Sequence::show ( ) [virtual]
```

Reimplemented from ModelElement.

Definition at line 21 of file Sequence.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Sequence.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Sequence.cpp

7.128 SequenceStep Class Reference

```
#include <Sequence.h>
```

Inheritance diagram for SequenceStep:

Collaboration diagram for SequenceStep:

Classes

- class Assignment

Public Member Functions

- SequenceStep (Station *station, std::list< Assignment * > *assignments=nullptr)
- virtual bool _loadInstance (std::map< std::string, std::string > *fields, unsigned int parentIndex)
- virtual std::map< std::string, std::string > * _saveInstance (unsigned int parentIndex)
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- std::list< SequenceStep::Assignment * > * getAssignments () const
- void setStation (Station *_station)
- Station * getStation () const

7.128.1 Detailed Description

Definition at line 22 of file Sequence.h.

7.128.2 Constructor & Destructor Documentation

7.128.2.1 SequenceStep()

```
SequenceStep::SequenceStep (
    Station * station,
    std::list< Assignment * > * assignments = nullptr )
```

Definition at line 75 of file Sequence.cpp.

7.128.3 Member Function Documentation

7.128.3.1 _loadInstance() [1/2]

```
bool SequenceStep::_loadInstance (
    std::map< std::string, std::string > * fields ) [virtual]
```

Implements PersistentObject_base.

Definition at line 117 of file Sequence.cpp.

7.128.3.2 _loadInstance() [2/2]

```
bool SequenceStep::_loadInstance (
    std::map< std::string, std::string > * fields,
    unsigned int parentIndex ) [virtual]
```

Definition at line 83 of file Sequence.cpp.

Here is the call graph for this function:

7.128.3.3 _saveInstance() [1/2]

```
std::map< std::string, std::string > * SequenceStep::_save↩
Instance ( ) [virtual]
```

Implements PersistentObject_base.

Definition at line 126 of file Sequence.cpp.

7.128.3.4 _saveInstance() [2/2]

```
std::map< std::string, std::string > * SequenceStep::_saveInstance (
            unsigned int parentIndex ) [virtual]
```

Definition at line 102 of file Sequence.cpp.

Here is the call graph for this function:

7.128.3.5 getAssignments()

```
std::list< SequenceStep::Assignment * > * SequenceStep::getAssignments ( ) const
```

Definition at line 130 of file Sequence.cpp.

Here is the caller graph for this function:

7.128.3.6 getStation()

```
Station * SequenceStep::getStation ( ) const
```

Definition at line 138 of file Sequence.cpp.

Here is the caller graph for this function:

7.128.3.7 setStation()

```
void SequenceStep::setStation (
    Station * _station )
```

Definition at line 134 of file Sequence.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Sequence.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Sequence.cpp

7.129 Set Class Reference

```
#include <Set.h>
```

Inheritance diagram for Set:

Collaboration diagram for Set:

Public Member Functions

- Set (Model ∗model, std::string name="")
- virtual ∼Set ()=default
- virtual std::string show ()
- void setSetOfType (std::string _setOfType)
- std::string getSetOfType () const
- List< ModelElement ∗ > ∗ getElementSet () const

Static Public Member Functions

- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)
- static PluginInformation ∗ GetPluginInformation ()

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual ParserChangesInformation ∗ _getParserChangesInformation ()

Additional Inherited Members

7.129.1 Detailed Description

Set module DESCRIPTION This data module defines various types of sets, including resource, counter, tally, entity type, and entity picture. Resource sets can be used in the Process modules (and Seize, Release, Enter, and Leave of the Advanced Process and Advanced Transfer panels). Counter and Tally sets can be used in the Record module. Queue sets can be used with the Seize, Hold, Access, Request, Leave, and Allocate modules of the Advanced Process and Advanced Transfer panels. TYPICAL USES Machines that can perform the same operations in a manufacturing facility Supervisors, check-out clerks in a store Shipping clerks, receptionists in an office Set of pictures corresponding to a set of entity types PROMPTS Prompt Description Name The unique name of the set being defined. Type Type of set being defined. Members Repeat group that specifies the resource members with the set. The order of listing the members within the repeat group is important when using selection rules such as Preferred Order and Cyclical. Resource Name Name of the resource to include in the resource set. Applies only when Type is Resource. Tally Name Name of the tally within the tally set. Applies only when Type is Tally. Counter Name Name of the counter within the counter set. Applies only when Type is Counter. Entity Type Name of the entity type within the entity type set. Applies only when Type is Entity. Picture Name Name of the picture within the picture set. Applies only when Type is Entity Picture.

Definition at line 56 of file Set.h.

7.129.2 Constructor & Destructor Documentation

7.129.2.1 Set()

```
Set::Set (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Set.cpp.

Here is the caller graph for this function:

7.129.2.2 ~Set()

```
virtual Set::~Set ( ) [virtual], [default]
```

7.129.3 Member Function Documentation

7.129.3.1 _check()

```
bool Set::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 85 of file Set.cpp.

Here is the call graph for this function:

7.129.3.2 _getParserChangesInformation()

```
ParserChangesInformation * Set::_getParserChanges↩
Information ( ) [protected], [virtual]
```

This method returns all changes in the parser that are needed by plugins of this ModelElements. When connecting a new plugin, ParserChangesInformation are used to change parser source code, whch is after compiled and dinamically linked to to simulator kernel to reflect the changes

Reimplemented from ModelElement.

Definition at line 100 of file Set.cpp.

7.129.3.3 _loadInstance()

```
bool Set::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 52 of file Set.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.129.3.4 _saveInstance() `std::map< std::string, std::string > * Set::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 73 of file Set.cpp.

Here is the call graph for this function:

7.129.3.5 getElementSet() `List< ModelElement * > * Set::getElementSet ( ) const`

Definition at line 33 of file Set.cpp.

Here is the caller graph for this function:

7.129.3.6 GetPluginInformation() `PluginInformation * Set::GetPluginInformation ( ) [static]`

Definition at line 37 of file Set.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.129.3.7 getSetOfType() `std::string Set::getSetOfType ( ) const`

Definition at line 29 of file Set.cpp.

7.129.3.8 LoadInstance() `ModelElement * Set::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 42 of file Set.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.129.3.9 setSetOfType() `void Set::setSetOfType ( std::string _setOfType )`

Definition at line 25 of file Set.cpp.

Here is the caller graph for this function:

7.129.3.10 show() `std::string Set::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 20 of file Set.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Set.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Set.cpp

7.130 Signal Class Reference

```
#include <Signal.h>
```

Inheritance diagram for Signal:

Collaboration diagram for Signal:

Public Member Functions

- Signal (Model ∗model, std::string name="")
- virtual ∼Signal ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.130.1 Detailed Description

Signal module DESCRIPTION The Signal module sends a signal value to each Hold module in the model set to Wait for Signal and releases the maximum specified number of entities. When an entity arrives at a Signal module, the signal is evaluated and the signal code is sent. At this time, entities at Hold modules that are waiting for the same signal are removed from their queues. The entity sending the signal continues processing until it encounters a delay, enters a queue, or is disposed. TYPICAL USES Analyzing traffic patterns at an intersection (signal when the light turns green) Signaling an operator to complete an order that was waiting for a component part PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Signal Value Value of the signal to be sent to entities in Hold modules. Limit Maximum number of entities that are to be released from any Hold modules when the signal is received.

Definition at line 38 of file Signal.h.

7.130.2 Constructor & Destructor Documentation

7.130.2.1 Signal()

```
Signal::Signal (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Signal.cpp.

Here is the caller graph for this function:

7.130.2.2 ~Signal()

```
virtual Signal::~Signal ( ) [virtual], [default]
```

7.130.3 Member Function Documentation

7.130.3.1 _check()

```
bool Signal::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Signal.cpp.

7.130.3.2 _execute()

```
void Signal::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Signal.cpp.

Here is the call graph for this function:

7.130.3.3 _initBetweenReplications()

```
void Signal::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 48 of file Signal.cpp.

7.130.3.4 _loadInstance()

```
bool Signal::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 40 of file Signal.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.130.3.5 _saveInstance() `std::map< std::string, std::string > * Signal::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Signal.cpp.

Here is the call graph for this function:

7.130.3.6 GetPluginInformation() `PluginInformation * Signal::GetPluginInformation ( ) [static]`

Definition at line 64 of file Signal.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.130.3.7 LoadInstance()

```
ModelComponent * Signal::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file Signal.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.130.3.8 show()

```
std::string Signal::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 21 of file Signal.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Signal.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Signal.cpp

7.131 SimulationControl Class Reference

```
#include <SimulationControl.h>
```

Inheritance diagram for SimulationControl:

Collaboration diagram for SimulationControl:

Public Member Functions

- SimulationControl (std::string type, std::string name, GetterMember getterMember, SetterMember setter↩ Member, std::string parent="")
- virtual ~SimulationControl ()=default
- std::string show ()
- void setValue (double value)

Additional Inherited Members

7.131.1 Detailed Description

Represents any possible parameter or control for a simulation. Any element or event the model can declare one of its own attribute as a simulation control. It just have to create a SimulationControl object, passing the access to the methods that gets and sets the control value and including this SimulationControl in the corresponding list of the model

Definition at line 24 of file SimulationControl.h.

7.131.2 Constructor & Destructor Documentation

7.131.2.1 SimulationControl()

```
SimulationControl::SimulationControl (
    std::string type,
    std::string name,
    GetterMember getterMember,
    SetterMember setterMember,
    std::string parent = "" )
```

Definition at line 18 of file SimulationControl.cpp.

7.131.2.2 ~SimulationControl()

```
virtual SimulationControl::~SimulationControl ( ) [virtual], [default]
```

7.131.3 Member Function Documentation

7.131.3.1 setValue()

```
void SimulationControl::setValue (
    double value )
```

Definition at line 26 of file SimulationControl.cpp.

7.131.3.2 show()

```
std::string SimulationControl::show ( )
```

Definition at line 22 of file SimulationControl.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationControl.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationControl.cpp

7.132 SimulationEvent Class Reference

```
#include <OnEventManager.h>
```

Collaboration diagram for SimulationEvent:

Public Member Functions

- void setSimulatedTime (double simulatedTime)
- double getSimulatedTime () const
- void setStopRequested (bool stopRequested)
- bool isStopRequested () const
- void setPauseRequested (bool pauseRequested)
- bool isPauseRequested () const
- void setPaused (bool Paused)
- bool isPaused () const
- void setRunning (bool Running)
- bool isRunning () const
- void setCustomObject (void ∗customObject)
- void ∗ getCustomObject () const
- void setCurrentReplicationNumber (unsigned int currentReplicationNumber)
- unsigned int getCurrentReplicationNumber () const
- void setCurrentEvent (Event ∗currentEvent)
- Event ∗ getCurrentEvent () const
- void setCurrentInputNumber (unsigned int currentInputNumber)
- unsigned int getCurrentInputNumber () const
- void setCurrentComponent (ModelComponent ∗currentComponent)
- ModelComponent ∗ getCurrentComponent () const
- void setCurrentEntity (Entity ∗currentEntity)
- Entity ∗ getCurrentEntity () const
- void setEntityCreated (Entity ∗entityCreated)
- Entity ∗ getEntityCreated () const
- void setDestinationComponent (ModelComponent ∗destinationComponent)
- ModelComponent ∗ getDestinationComponent () const

Friends

- class ModelSimulation

7.132.1 Detailed Description

Stores an event that happened on a specific replication

Definition at line 30 of file OnEventManager.h.

7.132.2 Member Function Documentation

7.132.2.1 getCurrentComponent() `ModelComponent* SimulationEvent::getCurrentComponent ( ) const [inline]`

Definition at line 89 of file OnEventManager.h.

7.132.2.2 getCurrentEntity() `Entity* SimulationEvent::getCurrentEntity ( ) const [inline]`

Definition at line 95 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.3 getCurrentEvent() `Event* SimulationEvent::getCurrentEvent ( ) const [inline]`

Definition at line 77 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.4 getCurrentInputNumber() `unsigned int SimulationEvent::getCurrentInputNumber ( ) const [inline]`

Definition at line 83 of file OnEventManager.h.

7.132.2.5 getCurrentReplicationNumber() `unsigned int SimulationEvent::getCurrentReplication↩ Number ( ) const [inline]`

Definition at line 71 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.6 getCustomObject() `void* SimulationEvent::getCustomObject ( ) const [inline]`

Definition at line 65 of file OnEventManager.h.

7.132.2.7 getDestinationComponent() `ModelComponent* SimulationEvent::getDestinationComponent ( ) const [inline]`

Definition at line 109 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.8 getEntityCreated() `Entity* SimulationEvent::getEntityCreated ( ) const [inline]`

Definition at line 101 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.9 getSimulatedTime() `double SimulationEvent::getSimulatedTime ( ) const [inline]`

Definition at line 35 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.10 isPaused() `bool SimulationEvent::isPaused ( ) const [inline]`

Definition at line 53 of file OnEventManager.h.

7.132.2.11 isPauseRequested() `bool SimulationEvent::isPauseRequested ( ) const [inline]`

Definition at line 47 of file OnEventManager.h.

7.132.2.12 isRunning() `bool SimulationEvent::isRunning ( ) const [inline]`

Definition at line 59 of file OnEventManager.h.

7.132.2.13 isStopRequested() `bool SimulationEvent::isStopRequested ( ) const [inline]`

Definition at line 41 of file OnEventManager.h.

7.132.2.14 setCurrentComponent() `void SimulationEvent::setCurrentComponent ( ModelComponent * currentComponent ) [inline]`

Definition at line 86 of file OnEventManager.h.

7.132.2.15 setCurrentEntity() `void SimulationEvent::setCurrentEntity ( Entity * currentEntity ) [inline]`

Definition at line 92 of file OnEventManager.h.

7.132.2.16 setCurrentEvent() `void SimulationEvent::setCurrentEvent ( Event * currentEvent ) [inline]`

Definition at line 74 of file OnEventManager.h.

7.132.2.17 setCurrentInputNumber() `void SimulationEvent::setCurrentInputNumber ( unsigned int currentInputNumber ) [inline]`

Definition at line 80 of file OnEventManager.h.

7.132.2.18 setCurrentReplicationNumber() `void SimulationEvent::setCurrentReplicationNumber ( unsigned int currentReplicationNumber ) [inline]`

Definition at line 68 of file OnEventManager.h.

7.132.2.19 setCustomObject()

```
void SimulationEvent::setCustomObject (
    void * customObject ) [inline]
```

Definition at line 62 of file OnEventManager.h.

7.132.2.20 setDestinationComponent()

```
void SimulationEvent::setDestinationComponent (
    ModelComponent * destinationComponent ) [inline]
```

Definition at line 105 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.21 setEntityCreated()

```
void SimulationEvent::setEntityCreated (
    Entity * entityCreated ) [inline]
```

Definition at line 98 of file OnEventManager.h.

Here is the caller graph for this function:

7.132.2.22 setPaused()

```
void SimulationEvent::setPaused (
    bool Paused ) [inline]
```

Definition at line 50 of file OnEventManager.h.

7.132.2.23 setPauseRequested()

```
void SimulationEvent::setPauseRequested (
    bool pauseRequested ) [inline]
```

Definition at line 44 of file OnEventManager.h.

7.132.2.24 setRunning() `void SimulationEvent::setRunning ( bool Running ) [inline]`

Definition at line 56 of file OnEventManager.h.

7.132.2.25 setSimulatedTime() `void SimulationEvent::setSimulatedTime ( double simulatedTime ) [inline]`

Definition at line 32 of file OnEventManager.h.

7.132.2.26 setStopRequested() `void SimulationEvent::setStopRequested ( bool stopRequested ) [inline]`

Definition at line 38 of file OnEventManager.h.

7.132.3 Friends And Related Function Documentation

7.132.3.1 ModelSimulation `friend class ModelSimulation [friend]`

Definition at line 115 of file OnEventManager.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/OnEventManager.h

7.133 SimulationExperiment Class Reference

`#include <SimulationExperiment.h>`

Collaboration diagram for SimulationExperiment:

Public Member Functions

- SimulationExperiment ()
- virtual ~SimulationExperiment ()=default

7.133.1 Detailed Description

Definition at line 17 of file SimulationExperiment.h.

7.133.2 Constructor & Destructor Documentation

7.133.2.1 SimulationExperiment() `SimulationExperiment::SimulationExperiment ( )`

Definition at line 16 of file SimulationExperiment.cpp.

7.133.2.2 ~SimulationExperiment() `virtual SimulationExperiment::~SimulationExperiment ( ) [virtual], [default]`

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationExperiment.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationExperiment.cpp

7.134 SimulationReporter_if Class Reference

```
#include <SimulationReporter_if.h>
```

Inheritance diagram for SimulationReporter_if:

Collaboration diagram for SimulationReporter_if:

Public Member Functions

- virtual void showReplicationStatistics ()=0
- virtual void showSimulationStatistics ()=0
- virtual void showSimulationResponses ()=0
- virtual void showSimulationControls ()=0

7.134.1 Detailed Description

Definition at line 20 of file SimulationReporter_if.h.

7.134.2 Member Function Documentation

7.134.2.1 showReplicationStatistics() `virtual void SimulationReporter_if::showReplicationStatistics ( ) [pure virtual]`

Implemented in SimulationReporterDefaultImpl1.

7.134.2.2 showSimulationControls() `virtual void SimulationReporter_if::showSimulationControls ( ) [pure virtual]`

Implemented in SimulationReporterDefaultImpl1.

7.134.2.3 showSimulationResponses() `virtual void SimulationReporter_if::showSimulation↩ Responses ( ) [pure virtual]`

Implemented in SimulationReporterDefaultImpl1.

7.134.2.4 showSimulationStatistics() `virtual void SimulationReporter_if::showSimulationStatistics ( ) [pure virtual]`

Implemented in SimulationReporterDefaultImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationReporter_if.h

7.135 SimulationReporterDefaultImpl1 Class Reference

```
#include <SimulationReporterDefaultImpl1.h>
```

Inheritance diagram for SimulationReporterDefaultImpl1:

Collaboration diagram for SimulationReporterDefaultImpl1:

Public Member Functions

- SimulationReporterDefaultImpl1 (ModelSimulation *simulation, Model *model, List< ModelElement * > *statsCountersSimulation)
- virtual ~SimulationReporterDefaultImpl1 ()=default
- virtual void showReplicationStatistics ()
- virtual void showSimulationStatistics ()
- virtual void showSimulationResponses ()
- virtual void showSimulationControls ()

7.135.1 Detailed Description

Class that implements SimulationReporter_if interface and is responsible for building and showing replication and simulation reports

Definition at line 26 of file SimulationReporterDefaultImpl1.h.

7.135.2 Constructor & Destructor Documentation

7.135.2.1 SimulationReporterDefaultImpl1()

```
SimulationReporterDefaultImpl1::SimulationReporter↩
DefaultImpl1 (
            ModelSimulation * simulation,
            Model * model,
            List< ModelElement * > * statsCountersSimulation )
```

Definition at line 22 of file SimulationReporterDefaultImpl1.cpp.

7.135.2.2 ~SimulationReporterDefaultImpl1()

```
virtual SimulationReporterDefaultImpl1::~Simulation↩
ReporterDefaultImpl1 ( ) [virtual], [default]
```

7.135.3 Member Function Documentation

7.135.3.1 showReplicationStatistics()

```
void SimulationReporterDefaultImpl1::showReplication↩
Statistics ( ) [virtual]
```

Implements SimulationReporter_if.

Definition at line 40 of file SimulationReporterDefaultImpl1.cpp.

7.135.3.2 showSimulationControls()

```
void SimulationReporterDefaultImpl1::showSimulationControls
( ) [virtual]
```

Implements SimulationReporter_if.

Definition at line 28 of file SimulationReporterDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.135.3.3 showSimulationResponses() `void SimulationReporterDefaultImpl1::showSimulation↩ Responses ( ) [virtual]`

Implements SimulationReporter_if.

Definition at line 152 of file SimulationReporterDefaultImpl1.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.135.3.4 showSimulationStatistics() `void SimulationReporterDefaultImpl1::showSimulationStatistics ( ) [virtual]`

TODO: USE REFERENCE TO MAPITEM TO AVOID COPY

Implements SimulationReporter_if.

Definition at line 164 of file SimulationReporterDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.cpp

7.136 SimulationResponse Class Reference

`#include <SimulationResponse.h>`

Inheritance diagram for SimulationResponse:

Collaboration diagram for SimulationResponse:

Public Member Functions

- SimulationResponse (std::string type, std::string name, GetterMember getterMember, std::string parent="")
- virtual ~SimulationResponse ()=default
- std::string show ()
- double getValue ()
- std::string getName () const
- void setName (std::string name)
- std::string getType () const
- std::string getParent () const

Protected Attributes

- std::string _type
- std::string _parent
- std::string _name
- GetterMember _getterMemberFunction

7.136.1 Detailed Description

Represents any possible response of a simulation. Any element or event the model can declare one of its own attribute as a simulation response. It just has to create a SimulationResponse object, passing the access to the method that gets the response value and including this SimulationResponse in the corresponding list of the model

Definition at line 25 of file SimulationResponse.h.

7.136.2 Constructor & Destructor Documentation

7.136.2.1 SimulationResponse()

```
SimulationResponse::SimulationResponse (
    std::string type,
    std::string name,
    GetterMember getterMember,
    std::string parent = "" )
```

Definition at line 19 of file SimulationResponse.cpp.

7.136.2.2 ~SimulationResponse()

```
virtual SimulationResponse::~SimulationResponse ( ) [virtual], [default]
```

7.136.3 Member Function Documentation

7.136.3.1 getName()

```
std::string SimulationResponse::getName ( ) const
```

Definition at line 30 of file SimulationResponse.cpp.

Here is the caller graph for this function:

7.136.3.2 getParent()

```
std::string SimulationResponse::getParent ( ) const
```

Definition at line 42 of file SimulationResponse.cpp.

7.136.3.3 getType() `std::string SimulationResponse::getType ( ) const`

Definition at line 38 of file SimulationResponse.cpp.

7.136.3.4 getValue() `double SimulationResponse::getValue ( )`

Definition at line 46 of file SimulationResponse.cpp.

Here is the caller graph for this function:

7.136.3.5 setName() `void SimulationResponse::setName ( std::string name )`

Definition at line 34 of file SimulationResponse.cpp.

7.136.3.6 show() `std::string SimulationResponse::show ( )`

Definition at line 26 of file SimulationResponse.cpp.

7.136.4 Member Data Documentation

7.136.4.1 _getterMemberFunction `GetterMember SimulationResponse::_getterMemberFunction [protected]`

Definition at line 41 of file SimulationResponse.h.

7.136.4.2 _name `std::string SimulationResponse::_name [protected]`

Definition at line 40 of file SimulationResponse.h.

7.136.4.3 _parent `std::string SimulationResponse::_parent [protected]`

Definition at line 39 of file SimulationResponse.h.

7.136.4.4 _type `std::string SimulationResponse::_type [protected]`

Definition at line 38 of file SimulationResponse.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationResponse.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationResponse.cpp

7.137 SimulationScenario Class Reference

```
#include <SimulationScenario.h>
```

Collaboration diagram for SimulationScenario:

Public Member Functions

- SimulationScenario ()
- virtual ~SimulationScenario ()=default
- bool startSimulation (Simulator *sim, std::string *errorMessage)
- std::list< std::pair< std::string, double > * > * getResponseValues () const
- std::list< std::pair< std::string, double > * > * getControlValues () const
- double getResponseValue (const std::string &responseName)
- void setModelFilename (std::string _modelFilename)
- std::string getModelFilename () const
- void setScenarioName (std::string _name)
- std::string getScenarioName () const
- void setScenarioDescription (std::string _scenarioDescription)
- std::string getScenarioDescription () const
- std::list< std::string > * getSelectedControls () const
- double getControlValue (const std::string &controlName)
- std::list< std::string > * getSelectedResponses () const
- void setSelectedControls (std::list< std::string > *selectedControls)
- void setControl (std::string name, double value) const

7.137.1 Detailed Description

Represents a scenario where a specific model (defined my ModelFilename) will be simulated. To each scenario will be associated a set of SimulationControl and SimulationResponse, and their values are set to the scenario by the ProcessAnalyser.

Definition at line 26 of file SimulationScenario.h.

7.137.2 Constructor & Destructor Documentation

7.137.2.1 SimulationScenario() `SimulationScenario::SimulationScenario ( ) [default]`

7.137.2.2 ~SimulationScenario() `virtual SimulationScenario::~SimulationScenario ( ) [virtual], [default]`

7.137.3 Member Function Documentation

7.137.3.1 getControlValue() `double SimulationScenario::getControlValue ( const std::string & controlName )`

Definition at line 74 of file SimulationScenario.cpp.

7.137.3.2 getControlValues() `std::list< std::pair< std::string, double > * > * SimulationScenario::getControlValues ( ) const`

Definition at line 61 of file SimulationScenario.cpp.

7.137.3.3 getModelFilename() `std::string SimulationScenario::getModelFilename ( ) const`

Definition at line 37 of file SimulationScenario.cpp.

7.137.3.4 getResponseValue() `double SimulationScenario::getResponseValue ( const std::string & responseName )`

Definition at line 65 of file SimulationScenario.cpp.

7.137.3.5 getResponseValues() `std::list< std::pair< std::string, double > * > * SimulationScenario::getResponseValues ( ) const`

The final result of the simulationScenario

Definition at line 57 of file SimulationScenario.cpp.

7.137.3.6 getScenarioDescription() `std::string SimulationScenario::getScenarioDescription ( ) const`

Definition at line 53 of file SimulationScenario.cpp.

7.137.3.7 getScenarioName() `std::string SimulationScenario::getScenarioName ( ) const`

Definition at line 29 of file SimulationScenario.cpp.

7.137.3.8 getSelectedControls() `std::list< std::string > * SimulationScenario::getSelectedControls ( ) const`

Definition at line 45 of file SimulationScenario.cpp.

7.137.3.9 getSelectedResponses() `std::list< std::string > * SimulationScenario::getSelected↩ Responses ( ) const`

Definition at line 41 of file SimulationScenario.cpp.

7.137.3.10 setControl() `void SimulationScenario::setControl ( std::string name, double value ) const`

Definition at line 83 of file SimulationScenario.cpp.

7.137.3.11 setModelFilename() `void SimulationScenario::setModelFilename ( std::string _modelFilename )`

Definition at line 33 of file SimulationScenario.cpp.

7.137.3.12 setScenarioDescription() `void SimulationScenario::setScenarioDescription ( std::string _scenarioDescription )`

Definition at line 49 of file SimulationScenario.cpp.

7.137.3.13 setScenarioName() `void SimulationScenario::setScenarioName ( std::string _name )`

Definition at line 25 of file SimulationScenario.cpp.

7.137.3.14 setSelectedControls() `void SimulationScenario::setSelectedControls ( std::list< std::string > * selectedControls )`

Definition at line 88 of file SimulationScenario.cpp.

7.137.3.15 startSimulation() `bool SimulationScenario::startSimulation ( Simulator * sim, std::string * errorMessage )`

Definition at line 18 of file SimulationScenario.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationScenario.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SimulationScenario.cpp

7.138 Simulator Class Reference

```
#include <Simulator.h>
```

Collaboration diagram for Simulator:

Public Member Functions

- Simulator ()
- virtual ~Simulator ()=default
- std::string getVersion () const
- unsigned int getVersionNumber () const
- std::string getName () const
- LicenceManager * getLicenceManager () const
- PluginManager * getPlugins () const
- ModelManager * getModels () const
- TraceManager * getTracer () const
- ParserManager * getParser () const
- ExperimentManager * getExperimenter () const

7.138.1 Detailed Description

The main class of the Genesys KERNEL simulation. It gives access to simulation models and tools. Simulation is the top level class and is supossed to be available to application as a dynamic linked library.

Definition at line 35 of file Simulator.h.

7.138.2 Constructor & Destructor Documentation

7.138.2.1 Simulator() `Simulator::Simulator ( )`

Definition at line 37 of file Simulator.cpp.

Here is the call graph for this function:

7.138.2.2 ~Simulator() `virtual Simulator::~Simulator ( ) [virtual], [default]`

Here is the caller graph for this function:

7.138.3 Member Function Documentation

7.138.3.1 getExperimenter() `ExperimentManager * Simulator::getExperimenter ( ) const`

Definition at line 66 of file Simulator.cpp.

7.138.3.2 getLicenceManager() `LicenceManager * Simulator::getLicenceManager ( ) const`

Definition at line 82 of file Simulator.cpp.

Here is the caller graph for this function:

7.138.3.3 getModels() `ModelManager * Simulator::getModels ( ) const`

Definition at line 54 of file Simulator.cpp.

Here is the caller graph for this function:

7.138.3.4 getName() `std::string Simulator::getName ( ) const`

Definition at line 78 of file Simulator.cpp.

7.138.3.5 getParser() `ParserManager * Simulator::getParser ( ) const`

Definition at line 62 of file Simulator.cpp.

7.138.3.6 getPlugins() `PluginManager * Simulator::getPlugins ( ) const`

Definition at line 50 of file Simulator.cpp.

Here is the caller graph for this function:

7.138.3.7 getTracer() `TraceManager * Simulator::getTracer ( ) const`

Definition at line 58 of file Simulator.cpp.

7.138.3.8 getVersion() `std::string Simulator::getVersion ( ) const`

Definition at line 70 of file Simulator.cpp.

Here is the caller graph for this function:

7.138.3.9 getVersionNumber() `unsigned int Simulator::getVersionNumber ( ) const`

Definition at line 74 of file Simulator.cpp.

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Simulator.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/Simulator.cpp

7.139 SinkModelComponent Class Reference

```
#include <SinkModelComponent.h>
```

Inheritance diagram for SinkModelComponent:

Collaboration diagram for SinkModelComponent:

Public Member Functions

- SinkModelComponent (Model *model, std::string componentTypename, std::string name="")
- virtual ~SinkModelComponent ()=default

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)

Additional Inherited Members

7.139.1 Detailed Description

This class is the basis for any component representing the end of a process flow, such as a Dispose. It can remove entities from the system and collect statistics.

Definition at line 25 of file SinkModelComponent.h.

7.139.2 Constructor & Destructor Documentation

7.139.2.1 SinkModelComponent()

```
SinkModelComponent::SinkModelComponent (
    Model * model,
    std::string componentTypename,
    std::string name = "" )
```

Definition at line 18 of file SinkModelComponent.cpp.

7.139.2.2 ~SinkModelComponent()

```
virtual SinkModelComponent::~SinkModelComponent ( ) [virtual], [default]
```

7.139.3 Member Function Documentation

7.139.3.1 _check()

```
bool SinkModelComponent::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Reimplemented in Dispose.

Definition at line 37 of file SinkModelComponent.cpp.

7.139.3.2 _initBetweenReplications()

```
void SinkModelComponent::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Reimplemented in Dispose.

Definition at line 29 of file SinkModelComponent.cpp.

Here is the caller graph for this function:

7.139.3.3 _loadInstance()

```
bool SinkModelComponent::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Reimplemented in Dispose.

Definition at line 21 of file SinkModelComponent.cpp.

Here is the call graph for this function:

7.139.3.4 _saveInstance()

```
std::map< std::string, std::string > * SinkModelComponent::_save↩
Instance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Reimplemented in Dispose.

Definition at line 32 of file SinkModelComponent.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SinkModelComponent.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SinkModelComponent.cpp

7.140 yy::genesyspp_parser::stack< T, S >::slice Class Reference

Present a slice of the top of a stack.

```
#include <GenesysParser.h>
```

Collaboration diagram for yy::genesyspp_parser::stack< T, S >::slice:

Public Member Functions

- slice (const stack &stack, index_type range)
- const T & operator[ ] (index_type i) const

7.140.1 Detailed Description

template<typename T, typename S = std::vector<T>>
class yy::genesyspp_parser::stack< T, S >::slice

Present a slice of the top of a stack.

Definition at line 2824 of file GenesysParser.h.

7.140.2 Constructor & Destructor Documentation

7.140.2.1 slice()

```
template<typename T , typename S = std::vector<T>>
yy::genesyspp_parser::stack< T, S >::slice::slice (
            const stack & stack,
            index_type range )  [inline]
```

Definition at line 2827 of file GenesysParser.h.

7.140.3 Member Function Documentation

7.140.3.1 operator[]()

```
template<typename T , typename S = std::vector<T>>
const T& yy::genesyspp_parser::stack< T, S >::slice::operator[] (
            index_type i ) const  [inline]
```

Definition at line 2833 of file GenesysParser.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.141 Smart_AssignWriteSeizes Class Reference

```
#include <Smart_AssignWriteSeizes.h>
```

Inheritance diagram for Smart_AssignWriteSeizes:

Collaboration diagram for Smart_AssignWriteSeizes:

Public Member Functions

- Smart_AssignWriteSeizes ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.141.1 Detailed Description

Definition at line 19 of file Smart_AssignWriteSeizes.h.

7.141.2 Constructor & Destructor Documentation

7.141.2.1 Smart_AssignWriteSeizes() `Smart_AssignWriteSeizes::Smart_AssignWriteSeizes ( )`

Definition at line 38 of file Smart_AssignWriteSeizes.cpp.

7.141.3 Member Function Documentation

7.141.3.1 main()

```
int Smart_AssignWriteSeizes::main (
    int argc,
    char ** argv ) [virtual]
```

Implements BaseConsoleGenesysApplication.

Definition at line 41 of file Smart_AssignWriteSeizes.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_AssignWriteSeizes.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_AssignWriteSeizes.cpp

7.142 Smart_Delay Class Reference

```
#include <Smart_Delay.h>
```

Inheritance diagram for Smart_Delay:

Collaboration diagram for Smart_Delay:

Public Member Functions

- Smart_Delay ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.142.1 Detailed Description

Definition at line 19 of file Smart_Delay.h.

7.142.2 Constructor & Destructor Documentation

7.142.2.1 Smart_Delay() `Smart_Delay::Smart_Delay ( )`

Definition at line 29 of file Smart_Delay.cpp.

7.142.3 Member Function Documentation

7.142.3.1 main() `int Smart_Delay::main ( int argc, char ** argv ) [virtual]`

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 36 of file Smart_Delay.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Delay.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Delay.cpp

7.143 Smart_Dummy Class Reference

```
#include <Smart_Dummy.h>
```

Inheritance diagram for Smart_Dummy:

Collaboration diagram for Smart_Dummy:

Public Member Functions

- Smart_Dummy ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.143.1 Detailed Description

Definition at line 19 of file Smart_Dummy.h.

7.143.2 Constructor & Destructor Documentation

7.143.2.1 Smart_Dummy() `Smart_Dummy::Smart_Dummy ( )`

Definition at line 29 of file Smart_Dummy.cpp.

7.143.3 Member Function Documentation

7.143.3.1 main()

```
int Smart_Dummy::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 36 of file Smart_Dummy.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Dummy.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Dummy.cpp

7.144 Smart_HoldSignal Class Reference

```
#include <Smart_HoldSignal.h>
```

Inheritance diagram for Smart_HoldSignal:

Collaboration diagram for Smart_HoldSignal:

Public Member Functions

- Smart_HoldSignal ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.144.1 Detailed Description

Definition at line 19 of file Smart_HoldSignal.h.

7.144.2 Constructor & Destructor Documentation

7.144.2.1 Smart_HoldSignal() `Smart_HoldSignal::Smart_HoldSignal ( )`

Definition at line 29 of file Smart_HoldSignal.cpp.

7.144.3 Member Function Documentation

7.144.3.1 main() `int Smart_HoldSignal::main ( int argc, char ** argv ) [virtual]`

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 36 of file Smart_HoldSignal.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_HoldSignal.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_HoldSignal.cpp

7.145 Smart_ModelInfoModelSimulation Class Reference

```
#include <Smart_ModelInfoModelSimulation.h>
```

Inheritance diagram for Smart_ModelInfoModelSimulation:

Collaboration diagram for Smart_ModelInfoModelSimulation:

Public Member Functions

- Smart_ModelInfoModelSimulation ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.145.1 Detailed Description

Definition at line 19 of file Smart_ModelInfoModelSimulation.h.

7.145.2 Constructor & Destructor Documentation

7.145.2.1 Smart_ModelInfoModelSimulation() `Smart_ModelInfoModelSimulation::Smart_ModelInfo↩ ModelSimulation ( )`

Definition at line 28 of file Smart_ModelInfoModelSimulation.cpp.

7.145.3 Member Function Documentation

7.145.3.1 main()

```
int Smart_ModelInfoModelSimulation::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 35 of file Smart_ModelInfoModelSimulation.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_ModelInfoModelSimulation.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_ModelInfoModelSimulation.cpp

7.146 Smart_OnEvent Class Reference

```
#include <Smart_OnEvent.h>
```

Inheritance diagram for Smart_OnEvent:

Collaboration diagram for Smart_OnEvent:

Public Member Functions

- Smart_OnEvent ()
- virtual int main (int argc, char **argv)
- void onBreakpointHandler (SimulationEvent *re)
- void onEntityCreateHandler (SimulationEvent *re)
- void onEntityMoveHandler (SimulationEvent *re)

- void onSimulationStartHandler (SimulationEvent *re)
- void onReplicationStartHandler (SimulationEvent *re)
- void onReplicationStepHandler (SimulationEvent *re)
- void onProcessEventHandler (SimulationEvent *re)
- void onReplicationEndHandler (SimulationEvent *re)
- void onSimulationEndHandler (SimulationEvent *re)
- void onSimulationPausedHandler (SimulationEvent *re)
- void onSimulationResumeHandler (SimulationEvent *re)
- void onEntityRemoveHandler (SimulationEvent *re)

Additional Inherited Members

7.146.1 Detailed Description

Definition at line 19 of file Smart_OnEvent.h.

7.146.2 Constructor & Destructor Documentation

7.146.2.1 Smart_OnEvent() `Smart_OnEvent::Smart_OnEvent ( )`

Definition at line 29 of file Smart_OnEvent.cpp.

7.146.3 Member Function Documentation

7.146.3.1 main()

```
int Smart_OnEvent::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 84 of file Smart_OnEvent.cpp.

7.146.3.2 onBreakpointHandler() `void Smart_OnEvent::onBreakpointHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 32 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.3 onEntityCreateHandler() `void Smart_OnEvent::onEntityCreateHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 36 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.4 onEntityMoveHandler() `void Smart_OnEvent::onEntityMoveHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 40 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.5 onEntityRemoveHandler() `void Smart_OnEvent::onEntityRemoveHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 44 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.6 onProcessEventHandler() `void Smart_OnEvent::onProcessEventHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 48 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.7 onReplicationEndHandler() `void Smart_OnEvent::onReplicationEndHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 52 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.8 onReplicationStartHandler()

```
void Smart_OnEvent::onReplicationStartHandler ( SimulationEvent * re ) [virtual]
```

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 56 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.9 onReplicationStepHandler()

```
void Smart_OnEvent::onReplicationStepHandler ( SimulationEvent * re ) [virtual]
```

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 60 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.10 onSimulationEndHandler() `void Smart_OnEvent::onSimulationEndHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 68 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.11 onSimulationPausedHandler() `void Smart_OnEvent::onSimulationPausedHandler ( SimulationEvent * re ) [virtual]`

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 72 of file Smart_OnEvent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.146.3.12 onSimulationResumeHandler()

```
void Smart_OnEvent::onSimulationResumeHandler ( SimulationEvent * re ) [virtual]
```

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 76 of file Smart_OnEvent.cpp.

Here is the caller graph for this function:

7.146.3.13 onSimulationStartHandler()

```
void Smart_OnEvent::onSimulationStartHandler ( SimulationEvent * re ) [virtual]
```

Reimplemented from BaseConsoleGenesysApplication.

Definition at line 64 of file Smart_OnEvent.cpp.

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_OnEvent.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_OnEvent.cpp

7.147 Smart_Parser Class Reference

```
#include <Smart_Parser.h>
```

Inheritance diagram for Smart_Parser:

Collaboration diagram for Smart_Parser:

Public Member Functions

- Smart_Parser ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.147.1 Detailed Description

Definition at line 19 of file Smart_Parser.h.

7.147.2 Constructor & Destructor Documentation

7.147.2.1 Smart_Parser() `Smart_Parser::Smart_Parser ( )`

Definition at line 28 of file Smart_Parser.cpp.

7.147.3 Member Function Documentation

7.147.3.1 main() `int Smart_Parser::main ( int argc, char ** argv ) [virtual]`

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 35 of file Smart_Parser.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Parser.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Parser.cpp

7.148 Smart_ParserModelFunctions Class Reference

```
#include <Smart_ParserModelFunctions.h>
```

Inheritance diagram for Smart_ParserModelFunctions:

Collaboration diagram for Smart_ParserModelFunctions:

Public Member Functions

- Smart_ParserModelFunctions ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.148.1 Detailed Description

Definition at line 19 of file Smart_ParserModelFunctions.h.

7.148.2 Constructor & Destructor Documentation

7.148.2.1 Smart_ParserModelFunctions() `Smart_ParserModelFunctions::Smart_ParserModelFunctions ( )`

Definition at line 33 of file Smart_ParserModelFunctions.cpp.

7.148.3 Member Function Documentation

7.148.3.1 main() `int Smart_ParserModelFunctions::main ( int argc, char ** argv ) [virtual]`

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 40 of file Smart_ParserModelFunctions.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.cpp

7.149 Smart_Process Class Reference

```
#include <Smart_Process.h>
```

Inheritance diagram for Smart_Process:

Collaboration diagram for Smart_Process:

Public Member Functions

- Smart_Process ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.149.1 Detailed Description

Definition at line 19 of file Smart_Process.h.

7.149.2 Constructor & Destructor Documentation

7.149.2.1 Smart_Process() `Smart_Process::Smart_Process ( )`

Definition at line 29 of file Smart_Process.cpp.

7.149.3 Member Function Documentation

7.149.3.1 main()

```
int Smart_Process::main (
    int argc,
    char ** argv )  [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 36 of file Smart_Process.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Process.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Process.cpp

7.150 Smart_ProcessSet Class Reference

```
#include <Smart_ProcessSet.h>
```

Inheritance diagram for Smart_ProcessSet:

Collaboration diagram for Smart_ProcessSet:

Public Member Functions

- Smart_ProcessSet ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.150.1 Detailed Description

Definition at line 19 of file Smart_ProcessSet.h.

7.150.2 Constructor & Destructor Documentation

7.150.2.1 Smart_ProcessSet() `Smart_ProcessSet::Smart_ProcessSet ( )`

Definition at line 29 of file Smart_ProcessSet.cpp.

7.150.3 Member Function Documentation

7.150.3.1 main()

```
int Smart_ProcessSet::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 36 of file Smart_ProcessSet.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_ProcessSet.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_ProcessSet.cpp

7.151 Smart_RouteStation Class Reference

```
#include <Smart_RouteStation.h>
```

Inheritance diagram for Smart_RouteStation:

Collaboration diagram for Smart_RouteStation:

Public Member Functions

- Smart_RouteStation ()
- virtual int main (int argc, char ∗∗argv)

Additional Inherited Members

7.151.1 Detailed Description

Definition at line 19 of file Smart_RouteStation.h.

7.151.2 Constructor & Destructor Documentation

7.151.2.1 Smart_RouteStation() `Smart_RouteStation::Smart_RouteStation ( )`

Definition at line 31 of file Smart_RouteStation.cpp.

7.151.3 Member Function Documentation

7.151.3.1 main()

```
int Smart_RouteStation::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 38 of file Smart_RouteStation.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_RouteStation.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_RouteStation.cpp

7.152 Smart_SeizeDelayRelease Class Reference

```
#include <Smart_SeizeDelayRelease.h>
```

Inheritance diagram for Smart_SeizeDelayRelease:

Collaboration diagram for Smart_SeizeDelayRelease:

Public Member Functions

- Smart_SeizeDelayRelease ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.152.1 Detailed Description

Definition at line 19 of file Smart_SeizeDelayRelease.h.

7.152.2 Constructor & Destructor Documentation

7.152.2.1 Smart_SeizeDelayRelease() `Smart_SeizeDelayRelease::Smart_SeizeDelayRelease ( )`

Definition at line 31 of file Smart_SeizeDelayRelease.cpp.

7.152.3 Member Function Documentation

7.152.3.1 main()

```
int Smart_SeizeDelayRelease::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 38 of file Smart_SeizeDelayRelease.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_SeizeDelayRelease.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_SeizeDelayRelease.cpp

7.153 Smart_SeizeDelayReleaseMany Class Reference

```
#include <Smart_SeizeDelayReleaseMany.h>
```

Inheritance diagram for Smart_SeizeDelayReleaseMany:

Collaboration diagram for Smart_SeizeDelayReleaseMany:

Public Member Functions

- Smart_SeizeDelayReleaseMany ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.153.1 Detailed Description

Definition at line 19 of file Smart_SeizeDelayReleaseMany.h.

7.153.2 Constructor & Destructor Documentation

7.153.2.1 Smart_SeizeDelayReleaseMany() `Smart_SeizeDelayReleaseMany::Smart_SeizeDelayRelease↩ Many ( )`

Definition at line 31 of file Smart_SeizeDelayReleaseMany.cpp.

7.153.3 Member Function Documentation

7.153.3.1 main()

```
int Smart_SeizeDelayReleaseMany::main (
    int argc,
    char ** argv ) [virtual]
```

This is the main function of the application. It instanciates the simulator, builds a simulation model and then simulate that model.

Implements BaseConsoleGenesysApplication.

Definition at line 38 of file Smart_SeizeDelayReleaseMany.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.cpp

7.154 Smart_Sequence Class Reference

```
#include <Smart_Sequence.h>
```

Inheritance diagram for Smart_Sequence:

Collaboration diagram for Smart_Sequence:

Public Member Functions

- Smart_Sequence ()
- virtual int main (int argc, char **argv)

Additional Inherited Members

7.154.1 Detailed Description

Definition at line 19 of file Smart_Sequence.h.

7.154.2 Constructor & Destructor Documentation

7.154.2.1 Smart_Sequence() `Smart_Sequence::Smart_Sequence ( )`

Definition at line 30 of file Smart_Sequence.cpp.

7.154.3 Member Function Documentation

7.154.3.1 main()

```
int Smart_Sequence::main (
    int argc,
    char ** argv )  [virtual]
```

Implements BaseConsoleGenesysApplication.

Definition at line 33 of file Smart_Sequence.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Sequence.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/examples/smarts/Smart_Sequence.cpp

7.155 Solver_if Class Reference

```
#include <solver_if.h>
```

Inheritance diagram for Solver_if:

Collaboration diagram for Solver_if:

Public Member Functions

- virtual void setPrecision (double e)=0
- virtual double getPrecision ()=0
- virtual void setMaxSteps (double steps)=0
- virtual double getMaxSteps ()=0
- virtual double integrate (double min, double max, double(∗f)(double, double), double p2)=0
- virtual double integrate (double min, double max, double(∗f)(double, double, double), double p2, double p3)=0
- virtual double integrate (double min, double max, double(∗f)(double, double, double, double), double p2, double p3, double p4)=0
- virtual double integrate (double min, double max, double(∗f)(double, double, double, double, double), double p2, double p3, double p4, double p5)=0
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double), double p2)=0
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double, double), double p2, double p3)=0
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double, double, double), double p2, double p3, double p4)=0
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double, double, double, double), double p2, double p3, double p4, double p5)=0

7.155.1 Detailed Description

Interface used by classes that perform the numerical integration and derivation of functions with from one up to four parameters. It is mainly used for calculating the probability of theoretical distributions, from its probability distribution functions. p1 is the value where function is being evaluated and p2, ... are the function parameters

Definition at line 22 of file solver_if.h.

7.155.2 Member Function Documentation

7.155.2.1 derivate() [1/4]

```
virtual double Solver_if::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double) f,
    double p2 )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.2 derivate() [2/4]

```
virtual double Solver_if::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double, double) f,
    double p2,
    double p3 )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.3 derivate() [3/4]

```
virtual double Solver_if::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double, double, double) f,
    double p2,
    double p3,
    double p4 )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.4 derivate() [4/4]

```
virtual double Solver_if::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double, double, double, double) f,
    double p2,
    double p3,
    double p4,
    double p5 )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.5 getMaxSteps() `virtual double Solver_if::getMaxSteps ( ) [pure virtual]`

Implemented in SolverDefaultImpl1.

Here is the caller graph for this function:

7.155.2.6 getPrecision() `virtual double Solver_if::getPrecision ( ) [pure virtual]`

Implemented in SolverDefaultImpl1.

7.155.2.7 integrate() [1/4]

```
virtual double Solver_if::integrate (
    double min,
    double max,
    double(*)(double, double) f,
    double p2 ) [pure virtual]
```

Implemented in SolverDefaultImpl1.

Here is the caller graph for this function:

7.155.2.8 integrate() [2/4]

```
virtual double Solver_if::integrate (
    double min,
    double max,
    double(*)(double, double, double) f,
    double p2,
    double p3 ) [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.9 integrate() [3/4]

```
virtual double Solver_if::integrate (
    double min,
    double max,
    double(*)(double, double, double, double) f,
    double p2,
    double p3,
    double p4 )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.10 integrate() [4/4]

```
virtual double Solver_if::integrate (
    double min,
    double max,
    double(*)(double, double, double, double, double) f,
    double p2,
    double p3,
    double p4,
    double p5 )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

7.155.2.11 setMaxSteps()

```
virtual void Solver_if::setMaxSteps (
    double steps )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

Here is the caller graph for this function:

7.155.2.12 setPrecision()

```
virtual void Solver_if::setPrecision (
    double e )  [pure virtual]
```

Implemented in SolverDefaultImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/solver_if.h

7.156 SolverDefaultImpl1 Class Reference

```
#include <SolverDefaultImpl1.h>
```

Inheritance diagram for SolverDefaultImpl1:

Collaboration diagram for SolverDefaultImpl1:

Public Member Functions

- SolverDefaultImpl1 (double precision=1e-6, unsigned int steps=1e3)
- virtual ~SolverDefaultImpl1 ()=default
- virtual void setPrecision (double e)
- virtual double getPrecision ()
- virtual void setMaxSteps (double steps)
- virtual double getMaxSteps ()
- virtual double integrate (double min, double max, double(*f)(double, double), double p2)

- virtual double integrate (double min, double max, double(∗f)(double, double, double), double p2, double p3)
- virtual double integrate (double min, double max, double(∗f)(double, double, double, double), double p2, double p3, double p4)
- virtual double integrate (double min, double max, double(∗f)(double, double, double, double, double), double p2, double p3, double p4, double p5)
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double), double p2)
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double, double), double p2, double p3)
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double, double, double), double p2, double p3, double p4)
- virtual double derivate (double initPoint, double initValue, double(∗f)(double, double, double, double, double), double p2, double p3, double p4, double p5)

7.156.1 Detailed Description

Definition at line 18 of file SolverDefaultImpl1.h.

7.156.2 Constructor & Destructor Documentation

7.156.2.1 SolverDefaultImpl1()

```
SolverDefaultImpl1::SolverDefaultImpl1 (
    double precision = 1e-6,
    unsigned int steps = 1e3 )
```

Definition at line 16 of file SolverDefaultImpl1.cpp.

7.156.2.2 ∼SolverDefaultImpl1()

```
virtual SolverDefaultImpl1::~SolverDefaultImpl1 ( ) [virtual], [default]
```

7.156.3 Member Function Documentation

7.156.3.1 derivate() [1/4]

```
double SolverDefaultImpl1::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double) f,
    double p2 ) [virtual]
```

Implements Solver_if.

Definition at line 130 of file SolverDefaultImpl1.cpp.

7.156.3.2 derivate() [2/4]

```
double SolverDefaultImpl1::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double, double) f,
    double p2,
    double p3 ) [virtual]
```

Implements Solver_if.

Definition at line 157 of file SolverDefaultImpl1.cpp.

7.156.3.3 derivate() [3/4]

```
double SolverDefaultImpl1::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double, double, double) f,
    double p2,
    double p3,
    double p4 ) [virtual]
```

Implements Solver_if.

Definition at line 160 of file SolverDefaultImpl1.cpp.

7.156.3.4 derivate() [4/4]

```
double SolverDefaultImpl1::derivate (
    double initPoint,
    double initValue,
    double(*)(double, double, double, double, double) f,
    double p2,
    double p3,
    double p4,
    double p5 ) [virtual]
```

Implements Solver_if.

Definition at line 163 of file SolverDefaultImpl1.cpp.

7.156.3.5 getMaxSteps()

```
double SolverDefaultImpl1::getMaxSteps ( ) [virtual]
```

Implements Solver_if.

Definition at line 33 of file SolverDefaultImpl1.cpp.

7.156.3.6 getPrecision()

```
double SolverDefaultImpl1::getPrecision ( ) [virtual]
```

Implements Solver_if.

Definition at line 25 of file SolverDefaultImpl1.cpp.

7.156.3.7 integrate() [1/4]

```
double SolverDefaultImpl1::integrate (
    double min,
    double max,
    double(*)(double, double) f,
    double p2 )  [virtual]
```

Implements Solver_if.

Definition at line 37 of file SolverDefaultImpl1.cpp.

7.156.3.8 integrate() [2/4]

```
double SolverDefaultImpl1::integrate (
    double min,
    double max,
    double(*)(double, double, double) f,
    double p2,
    double p3 )  [virtual]
```

Implements Solver_if.

Definition at line 60 of file SolverDefaultImpl1.cpp.

7.156.3.9 integrate() [3/4]

```
double SolverDefaultImpl1::integrate (
    double min,
    double max,
    double(*)(double, double, double, double) f,
    double p2,
    double p3,
    double p4 )  [virtual]
```

Implements Solver_if.

Definition at line 84 of file SolverDefaultImpl1.cpp.

7.156.3.10 integrate() [4/4]

```
double SolverDefaultImpl1::integrate (
    double min,
    double max,
    double(*)(double, double, double, double, double) f,
    double p2,
    double p3,
    double p4,
    double p5 )  [virtual]
```

Implements Solver_if.

Definition at line 107 of file SolverDefaultImpl1.cpp.

7.156.3.11 setMaxSteps() `void SolverDefaultImpl1::setMaxSteps ( double steps ) [virtual]`

Implements Solver_if.

Definition at line 29 of file SolverDefaultImpl1.cpp.

7.156.3.12 setPrecision() `void SolverDefaultImpl1::setPrecision ( double e ) [virtual]`

Implements Solver_if.

Definition at line 21 of file SolverDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/SolverDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/SolverDefaultImpl1.cpp

7.157 SortFile Class Reference

`#include <SorttFile.h>`

Collaboration diagram for SortFile:

Public Member Functions

- SortFile ()
- ~SortFile ()=default
- bool sort ()
- void setDataFilename (std::string filename)

7.157.1 Detailed Description

Definition at line 18 of file SorttFile.h.

7.157.2 Constructor & Destructor Documentation

7.157.2.1 SortFile() `SortFile::SortFile ( )`

Definition at line 18 of file SorttFile.cpp.

7.157.2.2 ~SortFile() `SortFile::~SortFile ( ) [default]`

7.157.3 Member Function Documentation

7.157.3.1 setDataFilename() `void SortFile::setDataFilename ( std::string filename )`

Definition at line 26 of file SorttFile.cpp.

7.157.3.2 sort() `bool SortFile::sort ( )`

Definition at line 21 of file SorttFile.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SorttFile.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/SorttFile.cpp

7.158 SourceModelComponent Class Reference

```
#include <SourceModelComponent.h>
```

Inheritance diagram for SourceModelComponent:

Collaboration diagram for SourceModelComponent:

Classes

- struct DEFAULT_VALUES

Public Member Functions

- SourceModelComponent (Model ∗model, std::string componentTypename, std::string name="")
- virtual ∼SourceModelComponent ()=default
- void setFirstCreation (double _firstCreation)
- double getFirstCreation () const
- void setEntityType (EntityType ∗_entityType)
- EntityType ∗ getEntityType () const
- void setTimeUnit (Util::TimeUnit _timeUnit)
- Util::TimeUnit getTimeUnit () const
- void setTimeBetweenCreationsExpression (std::string _timeBetweenCreations)
- std::string getTimeBetweenCreationsExpression () const
- void setMaxCreations (unsigned long _maxCreations)
- void setMaxCreations (std::string _maxCreationsExpression)
- std::string getMaxCreations () const
- unsigned int getEntitiesCreated () const
- void setEntitiesCreated (unsigned int _entitiesCreated)
- void setEntitiesPerCreation (unsigned int _entitiesPerCreation)
- unsigned int getEntitiesPerCreation () const
- virtual std::string show ()

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual void _initBetweenReplications ()
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Protected Attributes

- EntityType * _entityType
- const struct SourceModelComponent::DEFAULT_VALUES DEFAULT
- double _firstCreation = DEFAULT.firstCreation
- unsigned int _entitiesPerCreation = DEFAULT.entitiesPerCreation
- std::string _maxCreationsExpression = DEFAULT.maxCreationsExpression
- std::string _timeBetweenCreationsExpression = DEFAULT.timeBetweenCreationsExpression
- Util::TimeUnit _timeBetweenCreationsTimeUnit = DEFAULT.timeBetweenCreationsTimeUnit
- unsigned int _entitiesCreatedSoFar = 0

Additional Inherited Members

7.158.1 Detailed Description

A source component implements the base for inserting entities into the model when its simulation is initialized. During the initialization, the new and empty future events list is populated by events of creating entities and sending them to the source components existing in the model

Definition at line 27 of file SourceModelComponent.h.

7.158.2 Constructor & Destructor Documentation

7.158.2.1 SourceModelComponent()

```
SourceModelComponent::SourceModelComponent (
    Model * model,
    std::string componentTypename,
    std::string name = "" )
```

Definition at line 20 of file SourceModelComponent.cpp.

7.158.2.2 ~SourceModelComponent()

```
virtual SourceModelComponent::~SourceModelComponent ( ) [virtual], [default]
```

7.158.3 Member Function Documentation

7.158.3.1 _check() `bool SourceModelComponent::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Reimplemented in Create.

Definition at line 59 of file SourceModelComponent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.158.3.2 _initBetweenReplications() `void SourceModelComponent::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Reimplemented in Create.

Definition at line 44 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.3 _loadInstance()

`bool SourceModelComponent::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Reimplemented in Create.

Definition at line 30 of file SourceModelComponent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.158.3.4 _saveInstance()

`std::map< std::string, std::string > * SourceModelComponent::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Reimplemented in Create.

Definition at line 48 of file SourceModelComponent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.158.3.5 getEntitiesCreated() `unsigned int SourceModelComponent::getEntitiesCreated ( ) const`

Definition at line 129 of file SourceModelComponent.cpp.

7.158.3.6 getEntitiesPerCreation() `unsigned int SourceModelComponent::getEntitiesPerCreation ( ) const`

Definition at line 142 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.7 getEntityType() `EntityType * SourceModelComponent::getEntityType ( ) const`

Definition at line 97 of file SourceModelComponent.cpp.

7.158.3.8 getFirstCreation() `double SourceModelComponent::getFirstCreation ( ) const`

Definition at line 81 of file SourceModelComponent.cpp.

7.158.3.9 getMaxCreations() `std::string SourceModelComponent::getMaxCreations ( ) const`

Definition at line 125 of file SourceModelComponent.cpp.

7.158.3.10 getTimeBetweenCreationsExpression() `std::string SourceModelComponent::getTime↩ BetweenCreationsExpression ( ) const`

Definition at line 113 of file SourceModelComponent.cpp.

7.158.3.11 getTimeUnit() `Util::TimeUnit SourceModelComponent::getTimeUnit ( ) const`

Definition at line 105 of file SourceModelComponent.cpp.

7.158.3.12 setEntitiesCreated()

```
void SourceModelComponent::setEntitiesCreated (
    unsigned int _entitiesCreated )
```

Definition at line 133 of file SourceModelComponent.cpp.

7.158.3.13 setEntitiesPerCreation()

```
void SourceModelComponent::setEntitiesPerCreation (
    unsigned int _entitiesPerCreation )
```

Definition at line 137 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.14 setEntityType()

```
void SourceModelComponent::setEntityType (
    EntityType * _entityType )
```

Definition at line 93 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.15 setFirstCreation() `void SourceModelComponent::setFirstCreation ( double _firstCreation )`

Definition at line 77 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.16 setMaxCreations() [1/2]

```
void SourceModelComponent::setMaxCreations (
    std::string _maxCreationsExpression )
```

Definition at line 117 of file SourceModelComponent.cpp.

7.158.3.17 setMaxCreations() [2/2]

```
void SourceModelComponent::setMaxCreations (
    unsigned long _maxCreations )
```

Definition at line 121 of file SourceModelComponent.cpp.

7.158.3.18 setTimeBetweenCreationsExpression()

```
void SourceModelComponent::setTimeBetween↩
CreationsExpression (
    std::string _timeBetweenCreations )
```

Definition at line 109 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.19 setTimeUnit() `void SourceModelComponent::setTimeUnit ( Util::TimeUnit _timeUnit )`

Definition at line 101 of file SourceModelComponent.cpp.

Here is the caller graph for this function:

7.158.3.20 show() `std::string SourceModelComponent::show ( ) [virtual]`

Reimplemented from ModelComponent.

Reimplemented in Create.

Definition at line 23 of file SourceModelComponent.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.158.4 Member Data Documentation

7.158.4.1 _entitiesCreatedSoFar `unsigned int SourceModelComponent::_entitiesCreatedSoFar = 0 [protected]`

Definition at line 69 of file SourceModelComponent.h.

7.158.4.2 _entitiesPerCreation `unsigned int SourceModelComponent::_entitiesPerCreation = DEFAULT.↩ entitiesPerCreation [protected]`

Definition at line 65 of file SourceModelComponent.h.

7.158.4.3 _entityType `EntityType* SourceModelComponent::_entityType [protected]`

Definition at line 55 of file SourceModelComponent.h.

7.158.4.4 _firstCreation `double SourceModelComponent::_firstCreation = DEFAULT.firstCreation [protected]`

Definition at line 64 of file SourceModelComponent.h.

7.158.4.5 _maxCreationsExpression `std::string SourceModelComponent::_maxCreationsExpression = DEFAULT.maxCreationsExpression [protected]`

Definition at line 66 of file SourceModelComponent.h.

7.158.4.6 _timeBetweenCreationsExpression `std::string SourceModelComponent::_timeBetween↩ CreationsExpression = DEFAULT.timeBetweenCreationsExpression [protected]`

Definition at line 67 of file SourceModelComponent.h.

7.158.4.7 _timeBetweenCreationsTimeUnit `Util::TimeUnit SourceModelComponent::_timeBetween↩ CreationsTimeUnit = DEFAULT.timeBetweenCreationsTimeUnit [protected]`

Definition at line 68 of file SourceModelComponent.h.

7.158.4.8 DEFAULT `const struct SourceModelComponent::DEFAULT_VALUES SourceModelComponent::↩ DEFAULT [protected]`

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SourceModelComponent.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/SourceModelComponent.cpp

7.159 Start Class Reference

```
#include <Start.h>
```

Inheritance diagram for Start:

Collaboration diagram for Start:

Public Member Functions

- Start (Model *model, std::string name="")
- virtual ~Start ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelComponent * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual void _execute (Entity *entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)

Additional Inherited Members

7.159.1 Detailed Description

Start module DESCRIPTION The Start module changes the status of a conveyor from inactive to active. The conveyor may have been deactivated from either the Stop module or by initially being set to inactive at the start of the simulation. The velocity of the conveyor may be changed permanently when the conveyor is started. TYPICAL USES Start a bottling conveyor after scheduled maintenance Start a baggage claim conveyor when bags have arrived PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Conveyor Name Name of the conveyor to start. Velocity Speed of the conveyor once it begins to operate. This value will change the speed of the conveyor permanently, until it is changed in another module. Units Velocity time units.

Definition at line 39 of file Start.h.

7.159.2 Constructor & Destructor Documentation

7.159.2.1 Start()

```
Start::Start (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Start.cpp.

Here is the caller graph for this function:

7.159.2.2 ~Start()

```
virtual Start::~Start ( ) [virtual], [default]
```

7.159.3 Member Function Documentation

7.159.3.1 _check()

```
bool Start::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Start.cpp.

7.159.3.2 _execute()

```
void Start::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Start.cpp.

Here is the call graph for this function:

7.159.3.3 _initBetweenReplications() `void Start::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 48 of file Start.cpp.

7.159.3.4 _loadInstance() `bool Start::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 40 of file Start.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.159.3.5 _saveInstance() `std::map< std::string, std::string > * Start::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Start.cpp.

Here is the call graph for this function:

7.159.3.6 GetPluginInformation() `PluginInformation * Start::GetPluginInformation ( ) [static]`

Definition at line 64 of file Start.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.159.3.7 LoadInstance()

```
ModelComponent * Start::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file Start.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.159.3.8 show()

```
std::string Start::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 21 of file Start.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Start.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Start.cpp

7.160 Station Class Reference

```
#include <Station.h>
```

Inheritance diagram for Station:

Collaboration diagram for Station:

Public Member Functions

- Station (Model ∗model, std::string name="")
- virtual ∼Station ()
- virtual std::string show ()
- void initBetweenReplications ()
- void enter (Entity ∗entity)
- void leave (Entity ∗entity)
- void setEnterIntoStationComponent (ModelComponent ∗_enterIntoStationComponent)
- ModelComponent ∗ getEnterIntoStationComponent () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual void _createInternalElements ()

Additional Inherited Members

7.160.1 Detailed Description

Definition at line 64 of file Station.h.

7.160.2 Constructor & Destructor Documentation

7.160.2.1 Station()

```
Station::Station (
    Model * model,
    std::string name = "" )
```

Definition at line 19 of file Station.cpp.

Here is the caller graph for this function:

7.160.2.2 ~Station()

```
Station::~Station ( ) [virtual]
```

Definition at line 22 of file Station.cpp.

7.160.3 Member Function Documentation

7.160.3.1 _check()

```
bool Station::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 105 of file Station.cpp.

Here is the call graph for this function:

7.160.3.2 _createInternalElements() `void Station::_createInternalElements ( ) [protected], [virtual]`

This method is necessary only for those components that instantiate internal elements that must exist before simulation starts and even before model checking. That's the case of components that have internal StatisticsCollectors, since others components may refer to them as expressions (as in "TVAG(ThisCSTAT)") and therefore the element must exist before checking such expression

Reimplemented from ModelElement.

Definition at line 120 of file Station.cpp.

Here is the call graph for this function:

7.160.3.3 _loadInstance() `bool Station::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 90 of file Station.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.160.3.4 _saveInstance() `std::map< std::string, std::string > * Station::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 100 of file Station.cpp.

Here is the call graph for this function:

7.160.3.5 enter() `void Station::enter ( Entity * entity )`

Definition at line 42 of file Station.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.160.3.6 getEnterIntoStationComponent() `ModelComponent * Station::getEnterIntoStationComponent ( ) const`

Definition at line 71 of file Station.cpp.

Here is the caller graph for this function:

7.160.3.7 GetPluginInformation() `PluginInformation * Station::GetPluginInformation ( ) [static]`

Definition at line 75 of file Station.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.160.3.8 initBetweenReplications() `void Station::initBetweenReplications ( )`

Definition at line 36 of file Station.cpp.

Here is the call graph for this function:

7.160.3.9 leave() `void Station::leave ( Entity * entity )`

Definition at line 52 of file Station.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.160.3.10 LoadInstance()

```
ModelElement * Station::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 80 of file Station.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.160.3.11 setEnterIntoStationComponent()

```
void Station::setEnterIntoStationComponent (
    ModelComponent * _enterIntoStationComponent )
```

Definition at line 67 of file Station.cpp.

Here is the caller graph for this function:

7.160.3.12 show() `std::string Station::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 27 of file Station.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Station.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Station.cpp

7.161 Statistics_if Class Reference

```
#include <Statistics_if.h>
```

Inheritance diagram for Statistics_if:

Collaboration diagram for Statistics_if:

Public Member Functions

- virtual Collector_if ∗ getCollector ()=0
- virtual void setCollector (Collector_if ∗collector)=0
- virtual unsigned int numElements ()=0
- virtual double min ()=0
- virtual double max ()=0
- virtual double average ()=0
- virtual double variance ()=0
- virtual double stddeviation ()=0
- virtual double variationCoef ()=0
- virtual double halfWidthConfidenceInterval ()=0
- virtual unsigned int newSampleSize (double halfWidth)=0
- virtual double getConfidenceLevel ()=0
- virtual void setConfidenceLevel (double confidencelevel)=0

7.161.1 Detailed Description

Interface for statisct synthesis of a stochastic variable collected by a Collector_if. The statistics generated may be updated based only on the previous statistics and the single newest added value or they may be updated based on a datafile, depending on the Collector implementation.

Definition at line 23 of file Statistics_if.h.

7.161.2 Member Function Documentation

7.161.2.1 average() `virtual double Statistics_if::average ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.2 getCollector() `virtual Collector_if* Statistics_if::getCollector ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.3 getConfidenceLevel() `virtual double Statistics_if::getConfidenceLevel ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.4 halfWidthConfidenceInterval() `virtual double Statistics_if::halfWidthConfidence↩ Interval ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.5 max() `virtual double Statistics_if::max ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.6 min() `virtual double Statistics_if::min ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.7 newSampleSize() `virtual unsigned int Statistics_if::newSampleSize ( double halfWidth ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

7.161.2.8 numElements() `virtual unsigned int Statistics_if::numElements ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.9 setCollector()

`virtual void Statistics_if::setCollector ( Collector_if * collector ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

7.161.2.10 setConfidenceLevel()

`virtual void Statistics_if::setConfidenceLevel ( double confidencelevel ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

7.161.2.11 stddeviation()

`virtual double Statistics_if::stddeviation ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.12 variance() `virtual double Statistics_if::variance ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.161.2.13 variationCoef() `virtual double Statistics_if::variationCoef ( ) [pure virtual]`

Implemented in StatisticsDefaultImpl1, and StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/Statistics_if.h

7.162 StatisticsCollector Class Reference

```
#include <StatisticsCollector.h>
```

Inheritance diagram for StatisticsCollector:

Collaboration diagram for StatisticsCollector:

Public Member Functions

- StatisticsCollector (Model *model, std::string name="", ModelElement *parent=nullptr, bool insertInto↩ Model=true)
- virtual ~StatisticsCollector ()=default
- virtual std::string show ()
- ModelElement * getParent () const
- Statistics_if * getStatistics () const

Static Public Member Functions

- static PluginInformation * GetPluginInformation ()
- static ModelElement * LoadInstance (Model *model, std::map< std::string, std::string > *fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- void _addSimulationResponses ()

Additional Inherited Members

7.162.1 Detailed Description

The StatisticsCollector is the ModelElement responsible for collecting data from the model (using the Collector) and simultaneouly keeping statistics updated (using the Statistics)

Definition at line 27 of file StatisticsCollector.h.

7.162.2 Constructor & Destructor Documentation

7.162.2.1 StatisticsCollector()

```
StatisticsCollector::StatisticsCollector (
    Model * model,
    std::string name = "",
    ModelElement * parent = nullptr,
    bool insertIntoModel = true )
```

Definition at line 22 of file StatisticsCollector.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.162.2.2 ~StatisticsCollector()

```
virtual StatisticsCollector::~StatisticsCollector ( ) [virtual], [default]
```

Here is the caller graph for this function:

7.162.3 Member Function Documentation

7.162.3.1 _addSimulationResponses() `void StatisticsCollector::_addSimulationResponses ( ) [protected]`

Definition at line 28 of file StatisticsCollector.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.162.3.2 _check() `bool StatisticsCollector::_check ( std::string * errorMessage ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 100 of file StatisticsCollector.cpp.

7.162.3.3 _loadInstance()

```
bool StatisticsCollector::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 81 of file StatisticsCollector.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.162.3.4 _saveInstance()

```
std::map< std::string, std::string > * StatisticsCollector::_save↩
Instance ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 88 of file StatisticsCollector.cpp.

Here is the call graph for this function:

7.162.3.5 getParent() `ModelElement * StatisticsCollector::getParent ( ) const`

Definition at line 57 of file StatisticsCollector.cpp.

Here is the caller graph for this function:

7.162.3.6 GetPluginInformation() `PluginInformation * StatisticsCollector::GetPluginInformation ( ) [static]`

Definition at line 65 of file StatisticsCollector.cpp.

Here is the call graph for this function:

7.162.3.7 getStatistics() `Statistics_if * StatisticsCollector::getStatistics ( ) const`

Definition at line 61 of file StatisticsCollector.cpp.

Here is the caller graph for this function:

7.162.3.8 LoadInstance()

```
ModelElement * StatisticsCollector::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 71 of file StatisticsCollector.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.162.3.9 show() `std::string StatisticsCollector::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 43 of file StatisticsCollector.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/StatisticsCollector.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/StatisticsCollector.cpp

7.163 StatisticsDatafile_if Class Reference

```
#include <StatisticsDataFile_if.h>
```

Inheritance diagram for StatisticsDatafile_if:

Collaboration diagram for StatisticsDatafile_if:

Public Member Functions

- virtual double mode ()=0
- virtual double mediane ()=0
- virtual double quartil (unsigned short num)=0
- virtual double decil (unsigned short num)=0
- virtual double centil (unsigned short num)=0
- virtual void setHistogramNumClasses (unsigned short num)=0
- virtual unsigned short histogramNumClasses ()=0
- virtual double histogramClassLowerLimit (unsigned short classNum)=0
- virtual unsigned int histogramClassFrequency (unsigned short classNum)=0

7.163.1 Detailed Description

Definition at line 20 of file StatisticsDataFile_if.h.

7.163.2 Member Function Documentation

7.163.2.1 centil()

```
virtual double StatisticsDatafile_if::centil (
    unsigned short num ) [pure virtual]
```

Implemented in StatisticsDatafileDefaultImpl1.

7.163.2.2 decil()

```
virtual double StatisticsDatafile_if::decil (
    unsigned short num ) [pure virtual]
```

Implemented in StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.163.2.3 histogramClassFrequency()

```
virtual unsigned int StatisticsDatafile_if::histogram↩
ClassFrequency (
    unsigned short classNum ) [pure virtual]
```

Implemented in StatisticsDatafileDefaultImpl1.

7.163.2.4 histogramClassLowerLimit()

```
virtual double StatisticsDatafile_if::histogramClass↩
LowerLimit (
    unsigned short classNum ) [pure virtual]
```

Implemented in StatisticsDatafileDefaultImpl1.

7.163.2.5 histogramNumClasses() `virtual unsigned short StatisticsDatafile_if::histogramNum↩ Classes ( ) [pure virtual]`

Implemented in StatisticsDatafileDefaultImpl1.

7.163.2.6 mediane() `virtual double StatisticsDatafile_if::mediane ( ) [pure virtual]`

Implemented in StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.163.2.7 mode() `virtual double StatisticsDatafile_if::mode ( ) [pure virtual]`

Implemented in StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.163.2.8 quartil()

```
virtual double StatisticsDatafile_if::quartil (
    unsigned short num ) [pure virtual]
```

Implemented in StatisticsDatafileDefaultImpl1.

Here is the caller graph for this function:

7.163.2.9 setHistogramNumClasses()

```
virtual void StatisticsDatafile_if::setHistogramNumClasses
(
    unsigned short num ) [pure virtual]
```

Implemented in StatisticsDatafileDefaultImpl1.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/StatisticsDataFile_if.h

7.164 StatisticsDatafileDefaultImpl1 Class Reference

```
#include <StatisticsDataFileDefaultImpl.h>
```

Inheritance diagram for StatisticsDatafileDefaultImpl1:

Collaboration diagram for StatisticsDatafileDefaultImpl1:

Public Member Functions

- StatisticsDatafileDefaultImpl1 ()
- virtual ~StatisticsDatafileDefaultImpl1 ()=default
- virtual Collector_if * getCollector ()
- virtual void setCollector (Collector_if *collector)
- virtual unsigned int numElements ()

- virtual double min ()
- virtual double max ()
- virtual double average ()
- virtual double variance ()
- virtual double stddeviation ()
- virtual double variationCoef ()
- virtual double halfWidthConfidenceInterval ()
- virtual unsigned int newSampleSize (double halfWidth)
- virtual double getConfidenceLevel ()
- virtual void setConfidenceLevel (double confidencelevel)
- virtual double mode ()
- virtual double mediane ()
- virtual double quartil (unsigned short num)
- virtual double decil (unsigned short num)
- virtual double centil (unsigned short num)
- virtual void setHistogramNumClasses (unsigned short num)
- virtual unsigned short histogramNumClasses ()
- virtual double histogramClassLowerLimit (unsigned short classNum)
- virtual unsigned int histogramClassFrequency (unsigned short classNum)

7.164.1 Detailed Description

Definition at line 24 of file StatisticsDataFileDefaultImpl.h.

7.164.2 Constructor & Destructor Documentation

7.164.2.1 StatisticsDatafileDefaultImpl1() `StatisticsDatafileDefaultImpl1::StatisticsDatafile↩ DefaultImpl1 ( )`

Definition at line 18 of file StatisticsDataFileDefaultImpl.cpp.

7.164.2.2 ~StatisticsDatafileDefaultImpl1() `virtual StatisticsDatafileDefaultImpl1::~Statistics↩ DatafileDefaultImpl1 ( ) [virtual], [default]`

7.164.3 Member Function Documentation

7.164.3.1 average() `double StatisticsDatafileDefaultImpl1::average ( ) [virtual]`

Implements Statistics_if.

Definition at line 91 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.2 centil() `double StatisticsDatafileDefaultImpl1::centil ( unsigned short num ) [virtual]`

Implements StatisticsDatafile_if.

Definition at line 246 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.3 decil()

```
double StatisticsDatafileDefaultImpl1::decil (
    unsigned short num ) [virtual]
```

Implements StatisticsDatafile_if.

Definition at line 232 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.4 getCollector()

```
Collector_if * StatisticsDatafileDefaultImpl1::getCollector ( ) [virtual]
```

Implements Statistics_if.

Definition at line 31 of file StatisticsDataFileDefaultImpl.cpp.

7.164.3.5 getConfidenceLevel()

```
double StatisticsDatafileDefaultImpl1::getConfidenceLevel ( )
[virtual]
```

Implements Statistics_if.

Definition at line 161 of file StatisticsDataFileDefaultImpl.cpp.

7.164.3.6 halfWidthConfidenceInterval()

```
double StatisticsDatafileDefaultImpl1::halfWidthConfidence↩
Interval ( ) [virtual]
```

Implements Statistics_if.

Definition at line 140 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.7 histogramClassFrequency()

```
unsigned int StatisticsDatafileDefaultImpl1::histogram↩
ClassFrequency (
            unsigned short classNum )  [virtual]
```

Implements StatisticsDatafile_if.

Definition at line 284 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.8 histogramClassLowerLimit()

```
double StatisticsDatafileDefaultImpl1::histogramClass↩
LowerLimit (
            unsigned short classNum )  [virtual]
```

Implements StatisticsDatafile_if.

Definition at line 272 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.9 histogramNumClasses() `unsigned short StatisticsDatafileDefaultImpl1::histogramNum↩ Classes ( ) [virtual]`

Implements StatisticsDatafile_if.

Definition at line 264 of file StatisticsDataFileDefaultImpl.cpp.

Here is the caller graph for this function:

7.164.3.10 max() `double StatisticsDatafileDefaultImpl1::max ( ) [virtual]`

Implements Statistics_if.

Definition at line 81 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.11 mediane() `double StatisticsDatafileDefaultImpl1::mediane ( ) [virtual]`

Implements StatisticsDatafile_if.

Definition at line 196 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.12 min() `double StatisticsDatafileDefaultImpl1::min ( ) [virtual]`

Implements Statistics_if.

Definition at line 71 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.13 mode() `double StatisticsDatafileDefaultImpl1::mode ( ) [virtual]`

Implements StatisticsDatafile_if.

Definition at line 169 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.14 newSampleSize() `unsigned int StatisticsDatafileDefaultImpl1::newSampleSize ( double halfWidth ) [virtual]`

Implements Statistics_if.

Definition at line 150 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.15 numElements() `unsigned int StatisticsDatafileDefaultImpl1::numElements ( ) [virtual]`

Implements Statistics_if.

Definition at line 67 of file StatisticsDataFileDefaultImpl.cpp.

7.164.3.16 quartil()

```
double StatisticsDatafileDefaultImpl1::quartil (
    unsigned short num ) [virtual]
```

Implements StatisticsDatafile_if.

Definition at line 212 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

7.164.3.17 setCollector()

```
void StatisticsDatafileDefaultImpl1::setCollector (
    Collector_if * collector ) [virtual]
```

Implements Statistics_if.

Definition at line 35 of file StatisticsDataFileDefaultImpl.cpp.

7.164.3.18 setConfidenceLevel()

```
void StatisticsDatafileDefaultImpl1::setConfidenceLevel (
    double confidencelevel ) [virtual]
```

Implements Statistics_if.

Definition at line 165 of file StatisticsDataFileDefaultImpl.cpp.

7.164.3.19 setHistogramNumClasses()

```
void StatisticsDatafileDefaultImpl1::setHistogramNum↩
Classes (
    unsigned short num ) [virtual]
```

Implements StatisticsDatafile_if.

Definition at line 260 of file StatisticsDataFileDefaultImpl.cpp.

7.164.3.20 stddeviation() `double StatisticsDatafileDefaultImpl1::stddeviation ( ) [virtual]`

Implements Statistics_if.

Definition at line 115 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.21 variance() `double StatisticsDatafileDefaultImpl1::variance ( ) [virtual]`

Implements Statistics_if.

Definition at line 103 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.164.3.22 variationCoef() `double StatisticsDatafileDefaultImpl1::variationCoef ( ) [virtual]`

Implements Statistics_if.

Definition at line 123 of file StatisticsDataFileDefaultImpl.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.cpp

7.165 StatisticsDefaultImpl1 Class Reference

```
#include <StatisticsDefaultImpl1.h>
```

Inheritance diagram for StatisticsDefaultImpl1:

Collaboration diagram for StatisticsDefaultImpl1:

Public Member Functions

- StatisticsDefaultImpl1 ()

  When constructor is invoked without a Collector, it is taken from Traits<Statistics_if>::CollectorImplementation configuration.
- StatisticsDefaultImpl1 (Collector_if *collector)
- virtual ~StatisticsDefaultImpl1 ()=default
- virtual Collector_if * getCollector ()
- virtual void setCollector (Collector_if *collector)
- virtual unsigned int numElements ()
- virtual double min ()

- virtual double max ()
- virtual double average ()
- virtual double variance ()
- virtual double stddeviation ()
- virtual double variationCoef ()
- virtual double halfWidthConfidenceInterval ()
- virtual unsigned int newSampleSize (double halfWidth)
- virtual double getConfidenceLevel ()
- virtual void setConfidenceLevel (double confidencelevel)

7.165.1 Detailed Description

Definition at line 21 of file StatisticsDefaultImpl1.h.

7.165.2 Constructor & Destructor Documentation

7.165.2.1 StatisticsDefaultImpl1() [1/2] `StatisticsDefaultImpl1::StatisticsDefaultImpl1 ( )`

When constructor is invoked without a Collector, it is taken from Traits<Statistics_if>::CollectorImplementation configuration.

Definition at line 23 of file StatisticsDefaultImpl1.cpp.

Here is the call graph for this function:

7.165.2.2 StatisticsDefaultImpl1() [2/2] `StatisticsDefaultImpl1::StatisticsDefaultImpl1 ( Collector_if * collector )`

Definition at line 32 of file StatisticsDefaultImpl1.cpp.

Here is the call graph for this function:

7.165.2.3 ~StatisticsDefaultImpl1() `virtual StatisticsDefaultImpl1::~StatisticsDefaultImpl1 ( ) [virtual], [default]`

7.165.3 Member Function Documentation

7.165.3.1 average() `double StatisticsDefaultImpl1::average ( ) [virtual]`

Implements Statistics_if.

Definition at line 96 of file StatisticsDefaultImpl1.cpp.

7.165.3.2 getCollector() `Collector_if * StatisticsDefaultImpl1::getCollector ( ) [virtual]`

Implements Statistics_if.

Definition at line 131 of file StatisticsDefaultImpl1.cpp.

Here is the caller graph for this function:

7.165.3.3 getConfidenceLevel() `double StatisticsDefaultImpl1::getConfidenceLevel ( ) [virtual]`

Implements Statistics_if.

Definition at line 123 of file StatisticsDefaultImpl1.cpp.

7.165.3.4 halfWidthConfidenceInterval() `double StatisticsDefaultImpl1::halfWidthConfidence↩ Interval ( ) [virtual]`

Implements Statistics_if.

Definition at line 112 of file StatisticsDefaultImpl1.cpp.

7.165.3.5 max() `double StatisticsDefaultImpl1::max ( ) [virtual]`

Implements Statistics_if.

Definition at line 89 of file StatisticsDefaultImpl1.cpp.

7.165.3.6 min() `double StatisticsDefaultImpl1::min ( ) [virtual]`

Implements Statistics_if.

Definition at line 82 of file StatisticsDefaultImpl1.cpp.

7.165.3.7 newSampleSize() `unsigned int StatisticsDefaultImpl1::newSampleSize ( double halfWidth ) [virtual]`

Implements Statistics_if.

Definition at line 127 of file StatisticsDefaultImpl1.cpp.

7.165.3.8 numElements() `unsigned int StatisticsDefaultImpl1::numElements ( ) [virtual]`

Implements Statistics_if.

Definition at line 78 of file StatisticsDefaultImpl1.cpp.

Here is the call graph for this function:

7.165.3.9 setCollector() void StatisticsDefaultImpl1::setCollector (Collector_if * *collector*) [virtual]

Implements Statistics_if.

Definition at line 135 of file StatisticsDefaultImpl1.cpp.

7.165.3.10 setConfidenceLevel() void StatisticsDefaultImpl1::setConfidenceLevel (double *confidencelevel*) [virtual]

Implements Statistics_if.

Definition at line 116 of file StatisticsDefaultImpl1.cpp.

7.165.3.11 stddeviation() double StatisticsDefaultImpl1::stddeviation ( ) [virtual]

Implements Statistics_if.

Definition at line 104 of file StatisticsDefaultImpl1.cpp.

7.165.3.12 variance() double StatisticsDefaultImpl1::variance ( ) [virtual]

Implements Statistics_if.

Definition at line 100 of file StatisticsDefaultImpl1.cpp.

7.165.3.13 variationCoef() double StatisticsDefaultImpl1::variationCoef ( ) [virtual]

Implements Statistics_if.

Definition at line 108 of file StatisticsDefaultImpl1.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/StatisticsDefaultImpl1.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/statistics/StatisticsDefaultImpl1.cpp

7.166 Stop Class Reference

```
#include <Stop.h>
```

Inheritance diagram for Stop:

Collaboration diagram for Stop:

Public Member Functions

- Stop (Model ∗model, std::string name="")
- virtual ∼Stop ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.166.1 Detailed Description

Stop module DESCRIPTION The Stop module sets the operational status of a conveyor to inactive. The conveyor may have been activated from either the Start module or by initially being set to active at the start of the simulation. When the entity enters the Stop module, the conveyor will stop immediately, regardless of the type of conveyor or the number of entities currently on the conveyor. TYPICAL USES Stop a baggage conveyor after a pre-determined amount of time Stop a conveyor for scheduled maintenance PROMPTS Prompt Description Name Unique name of the module that will be displayed in the flowchart. Conveyor Name Name of the conveyor to stop.

Definition at line 36 of file Stop.h.

7.166.2 Constructor & Destructor Documentation

7.166.2.1 Stop()

```
Stop::Stop (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Stop.cpp.

Here is the caller graph for this function:

7.166.2.2 ~Stop()

```
virtual Stop::~Stop ( ) [virtual], [default]
```

7.166.3 Member Function Documentation

7.166.3.1 _check()

```
bool Stop::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Stop.cpp.

7.166.3.2 _execute()

```
void Stop::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Stop.cpp.

Here is the call graph for this function:

7.166.3.3 _initBetweenReplications() `void Stop::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 48 of file Stop.cpp.

7.166.3.4 _loadInstance() `bool Stop::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 40 of file Stop.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.166.3.5 _saveInstance() `std::map< std::string, std::string > * Stop::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Stop.cpp.

Here is the call graph for this function:

7.166.3.6 GetPluginInformation() `PluginInformation * Stop::GetPluginInformation ( ) [static]`

Definition at line 64 of file Stop.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.166.3.7 LoadInstance()

```
ModelComponent * Stop::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 25 of file Stop.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.166.3.8 show()

```
std::string Stop::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 21 of file Stop.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Stop.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Stop.cpp

7.167 Storage Class Reference

```
#include <Storage.h>
```

Inheritance diagram for Storage:

Collaboration diagram for Storage:

Public Member Functions

- Storage (Model ∗model, std::string name="")
- virtual ∼Storage ()=default
- virtual std::string show ()

Static Public Member Functions

- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)
- static PluginInformation ∗ GetPluginInformation ()

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)
- virtual ParserChangesInformation ∗ _getParserChangesInformation ()

Additional Inherited Members

7.167.1 Detailed Description

Definition at line 37 of file Storage.h.

7.167.2 Constructor & Destructor Documentation

7.167.2.1 Storage()

```
Storage::Storage (
    Model * model,
    std::string name = "" )
```

Definition at line 16 of file Storage.cpp.

Here is the caller graph for this function:

7.167.2.2 ~Storage()

```
virtual Storage::~Storage ( ) [virtual], [default]
```

7.167.3 Member Function Documentation

7.167.3.1 _check()

```
bool Storage::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 58 of file Storage.cpp.

7.167.3.2 _getParserChangesInformation()

```
ParserChangesInformation * Storage::_getParserChanges↩
Information ( ) [protected], [virtual]
```

This method returns all changes in the parser that are needed by plugins of this ModelElements. When connecting a new plugin, ParserChangesInformation are used to change parser source code, whch is after compiled and dinamically linked to to simulator kernel to reflect the changes

Reimplemented from ModelElement.

Definition at line 65 of file Storage.cpp.

7.167.3.3 _loadInstance()

```
bool Storage::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 39 of file Storage.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.167.3.4 _saveInstance()

```
std::map< std::string, std::string > * Storage::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 51 of file Storage.cpp.

Here is the call graph for this function:

7.167.3.5 GetPluginInformation() `PluginInformation * Storage::GetPluginInformation ( ) [static]`

Definition at line 24 of file Storage.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.167.3.6 LoadInstance() `ModelElement * Storage::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 29 of file Storage.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.167.3.7 show() `std::string Storage::show ( ) [virtual]`

Reimplemented from ModelElement.

Definition at line 19 of file Storage.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Storage.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Storage.cpp

7.168 Store Class Reference

```
#include <Store.h>
```

Inheritance diagram for Store:

Collaboration diagram for Store:

Public Member Functions

- Store (Model ∗model, std::string name="")
- virtual ∼Store ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.168.1 Detailed Description

Store module DESCRIPTION The Store module adds an entity to storage. The Unstore module may then be used to remove the entity from the storage. When an entity arrives at the Store module, the storage specified is incremented, and the entity immediately moves to the next module in the model. Storages are useful for displaying entity animation while an entity undergoes processing in other modules. Additionally, statistics may be kept on the number of entities in storage. TYPICAL USES Animating a part through a number of delay operations (load, setup, process, unload) Tracking the number of customers within a grocery store (place in storage upon entry) PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Method of specifying the storage name as a Storage, Set, Attribute, or Expression. Storage Name Name of the storage to which the entity will be added. Applies only when the Type is Storage. Set Name Name of the storage set from which the storage is to be selected. Applies only when the Type is Set. Set Index Index into the defined storage set that contains the desired storage name. Applies only when the Type is Set. Attribute Name of the attribute whose value contains the storage. Applies only when the Type is Attribute. Expression Expression that is evaluated to the storage into which the entity is placed. Applies only when the Type is Expression.

Definition at line 50 of file Store.h.

7.168.2 Constructor & Destructor Documentation

7.168.2.1 Store()

```
Store::Store (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Store.cpp.

Here is the caller graph for this function:

7.168.2.2 ~Store()

```
virtual Store::~Store ( ) [virtual], [default]
```

7.168.3 Member Function Documentation

7.168.3.1 _check()

```
bool Store::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file Store.cpp.

7.168.3.2 _execute()

```
void Store::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file Store.cpp.

Here is the call graph for this function:

7.168.3.3 _initBetweenReplications()

```
void Store::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 47 of file Store.cpp.

7.168.3.4 _loadInstance()

```
bool Store::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 39 of file Store.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.168.3.5 _saveInstance()

```
std::map< std::string, std::string > * Store::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 50 of file Store.cpp.

Here is the call graph for this function:

7.168.3.6 GetPluginInformation() `PluginInformation * Store::GetPluginInformation ( ) [static]`

Definition at line 63 of file Store.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.168.3.7 LoadInstance() `ModelComponent * Store::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 24 of file Store.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.168.3.8 show() `std::string Store::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 20 of file Store.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Store.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Store.cpp

7.169 Submodel Class Reference

```
#include <Submodel.h>
```

Inheritance diagram for Submodel:

Collaboration diagram for Submodel:

Public Member Functions

- Submodel (Model ∗model, std::string name="")
- virtual ∼Submodel ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.169.1 Detailed Description

This component ...

Definition at line 22 of file Submodel.h.

7.169.2 Constructor & Destructor Documentation

7.169.2.1 Submodel()

```
Submodel::Submodel (
    Model * model,
    std::string name = "" )
```

Definition at line 18 of file Submodel.cpp.

Here is the caller graph for this function:

7.169.2.2 ~Submodel()

```
virtual Submodel::~Submodel ( ) [virtual], [default]
```

7.169.3 Member Function Documentation

7.169.3.1 _check()

```
bool Submodel::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 57 of file Submodel.cpp.

7.169.3.2 _execute()

```
void Submodel::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 35 of file Submodel.cpp.

Here is the call graph for this function:

7.169.3.3 _initBetweenReplications()

`void Submodel::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 48 of file Submodel.cpp.

7.169.3.4 _loadInstance()

`bool Submodel::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 40 of file Submodel.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.169.3.5 _saveInstance() `std::map< std::string, std::string > * Submodel::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 51 of file Submodel.cpp.

Here is the call graph for this function:

7.169.3.6 GetPluginInformation() `PluginInformation * Submodel::GetPluginInformation ( ) [static]`

Definition at line 64 of file Submodel.cpp.

Here is the call graph for this function:

7.169.3.7 LoadInstance() `ModelComponent * Submodel::LoadInstance ( Model * model, std::map< std::string, std::string > * fields ) [static]`

Definition at line 25 of file Submodel.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.169.3.8 show() `std::string Submodel::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 21 of file Submodel.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Submodel.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Submodel.cpp

7.170 yy::genesyspp_parser::symbol_kind Struct Reference

Symbol kinds.

```
#include <GenesysParser.h>
```

Collaboration diagram for yy::genesyspp_parser::symbol_kind:

Public Types

- enum symbol_kind_type {
  YYNTOKENS = 90 , S_YYEMPTY = -2 , S_YYEOF = 0 , S_YYerror = 1 ,
  S_YYUNDEF = 2 , S_NUMD = 3 , S_NUMH = 4 , S_CTEZERO = 5 ,
  S_oLE = 6 , S_oGE = 7 , S_oEQ = 8 , S_oNE = 9 ,
  S_oAND = 10 , S_oOR = 11 , S_oNAND = 12 , S_oXOR = 13 ,
  S_oNOT = 14 , S_fSIN = 15 , S_fCOS = 16 , S_fROUND = 17 ,
  S_fMOD = 18 , S_fTRUNC = 19 , S_fFRAC = 20 , S_fEXP = 21 ,
  S_fSQRT = 22 , S_fLOG = 23 , S_fLN = 24 , S_fVAL = 25 ,
  S_fEVAL = 26 , S_fLENG = 27 , S_fRND1 = 28 , S_fEXPO = 29 ,
  S_fNORM = 30 , S_fUNIF = 31 , S_fWEIB = 32 , S_fLOGN = 33 ,
  S_fGAMM = 34 , S_fERLA = 35 , S_fTRIA = 36 , S_fBETA = 37 ,
  S_fDISC = 38 , S_fTNOW = 39 , S_fTFIN = 40 , S_fMAXREP = 41 ,
  S_fNUMREP = 42 , S_fIDENT = 43 , S_cIF = 44 , S_cELSE = 45 ,
  S_cFOR = 46 , S_cTO = 47 , S_cDO = 48 , S_ATRIB = 49 ,
  S_CSTAT = 50 , S_fTAVG = 51 , S_ILLEGAL = 52 , S_RESOURCE = 53 ,
  S_fNR = 54 , S_fMR = 55 , S_fIRF = 56 , S_fRESSEIZES = 57 ,
  S_fSTATE = 58 , S_fSETSUM = 59 , S_fRESUTIL = 60 , S_QUEUE = 61 ,
  S_fNQ = 62 , S_fFIRSTINQ = 63 , S_fLASTINQ = 64 , S_fSAQUE = 65 ,
  S_fAQUE = 66 , S_fENTATRANK = 67 , S_SET = 68 , S_fNUMSET = 69 ,
  S_VARI = 70 , S_FORM = 71 , S_fNUMGR = 72 , S_fATRGR = 73 ,
  S_LPAREN = 74 , S_RPAREN = 75 , S_LBRACKET = 76 , S_RBRACKET = 77 ,
  S_PLUS = 78 , S_MINUS = 79 , S_STAR = 80 , S_POWER = 81 ,
  S_SLASH = 82 , S_LESS = 83 , S_GREATER = 84 , S_ASSIGN = 85 ,
  S_COMMA = 86 , S_NEG = 87 , S_88_n_ = 88 , S_89_USER_ = 89 ,
  S_YYACCEPT = 90 , S_input = 91 , S_expression = 92 , S_numero = 93 ,
  S_aritmetica = 94 , S_logica = 95 , S_relacional = 96 , S_comando = 97 ,
  S_comandoIF = 98 , S_comandoFOR = 99 , S_function = 100 , S_kernelFunction = 101 ,
  S_trigonFunction = 102 , S_mathFunction = 103 , S_probFunction = 104 , S_userFunction = 105 ,
  S_listaparm = 106 , S_atributo = 107 , S_variavel = 108 , S_formula = 109 ,
  S_atribuicao = 110 , S_pluginFunction = 111 }

7.170.1 Detailed Description

Symbol kinds.

Definition at line 671 of file GenesysParser.h.

7.170.2 Member Enumeration Documentation

7.170.2.1 symbol_kind_type `enum yy::genesyspp_parser::symbol_kind::symbol_kind_type`

Enumerator

YYNTOKENS	Number of tokens.
S_YYEMPTY	
S_YYEOF	
S_YYerror	
S_YYUNDEF	
S_NUMD	

Enumerator

S_NUMH		
S_CTEZERO		
S_oLE		
S_oGE		
S_oEQ		
S_oNE		
S_oAND		
S_oOR		
S_oNAND		
S_oXOR		
S_oNOT		
S_fSIN		
S_fCOS		
S_fROUND		
S_fMOD		
S_fTRUNC		
S_fFRAC		
S_fEXP		
S_fSQRT		
S_fLOG		
S_fLN		
S_fVAL		
S_fEVAL		
S_fLENG		
S_fRND1		
S_fEXPO		
S_fNORM		
S_fUNIF		
S_fWEIB		
S_fLOGN		
S_fGAMM		
S_fERLA		
S_fTRIA		
S_fBETA		
S_fDISC		
S_fTNOW		
S_fTFIN		
S_fMAXREP		
S_fNUMREP		
S_fIDENT		
S_cIF		
S_cELSE		
S_cFOR		
S_cTO		
S_cDO		
S_ATRIB		
S_CSTAT		
S_fTAVG		
S_ILLEGAL		

Enumerator

S_RESOURCE		
S_fNR		
S_fMR		
S_fIRF		
S_fRESSEIZES		
S_fSTATE		
S_fSETSUM		
S_fRESUTIL		
S_QUEUE		
S_fNQ		
S_fFIRSTINQ		
S_fLASTINQ		
S_fSAQUE		
S_fAQUE		
S_fENTATRANK		
S_SET		
S_fNUMSET		
S_VARI		
S_FORM		
S_fNUMGR		
S_fATRGR		
S_LPAREN		
S_RPAREN		
S_LBRACKET		
S_RBRACKET		
S_PLUS		
S_MINUS		
S_STAR		
S_POWER		
S_SLASH		
S_LESS		
S_GREATER		
S_ASSIGN		
S_COMMA		
S_NEG		
S_88_n_		
S_89_USER_		
S_YYACCEPT		
S_input		
S_expression		
S_numero		
S_aritmetica		
S_logica		
S_relacional		
S_comando		
S_comandoIF		
S_comandoFOR		
S_function		
S_kernelFunction		

Enumerator

S_trigonFunction		
S_mathFunction		
S_probFunction		
S_userFunction		
S_listaparm		
S_atributo		
S_variavel		
S_formula		
S_atribuicao		
S_pluginFunction		

Definition at line 673 of file GenesysParser.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.171 yy::genesyspp_parser::symbol_type Struct Reference

"External" symbols: returned by the scanner.

```
#include <GenesysParser.h>
```

Inheritance diagram for yy::genesyspp_parser::symbol_type:

Collaboration diagram for yy::genesyspp_parser::symbol_type:

Public Types

- typedef basic_symbol< by_kind > super_type

  Superclass.

Public Member Functions

- symbol_type ()

Empty symbol.

- symbol_type (int tok, const location_type &l)

  Constructor for valueless symbols, and symbols from each type.

- symbol_type (int tok, const obj_t &v, const location_type &l)

Additional Inherited Members

7.171.1 Detailed Description

"External" symbols: returned by the scanner.

Definition at line 1151 of file GenesysParser.h.

7.171.2 Member Typedef Documentation

7.171.2.1 super_type `typedef basic_symbol<by_kind> yy::genesyspp_parser::symbol_type::super_type`

Superclass.

Definition at line 1154 of file GenesysParser.h.

7.171.3 Constructor & Destructor Documentation

7.171.3.1 symbol_type() [1/3] `yy::genesyspp_parser::symbol_type::symbol_type ( ) [inline]`

Empty symbol.

Definition at line 1157 of file GenesysParser.h.

7.171.3.2 symbol_type() [2/3]

```
yy::genesyspp_parser::symbol_type::symbol_type (
    int tok,
    const location_type & l ) [inline]
```

Constructor for valueless symbols, and symbols from each type.

Definition at line 1164 of file GenesysParser.h.

7.171.3.3 symbol_type() [3/3]

```
yy::genesyspp_parser::symbol_type::symbol_type (
    int tok,
    const obj_t & v,
    const location_type & l ) [inline]
```

Definition at line 1178 of file GenesysParser.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.172 yy::genesyspp_parser::syntax_error Struct Reference

Syntax errors thrown from user actions.

```
#include <GenesysParser.h>
```

Inheritance diagram for yy::genesyspp_parser::syntax_error:

Collaboration diagram for yy::genesyspp_parser::syntax_error:

Public Member Functions

- syntax_error (const location_type &l, const std::string &m)
- syntax_error (const syntax_error &s)
- ~syntax_error () YY_NOEXCEPT YY_NOTHROW

Public Attributes

- location_type location

7.172.1 Detailed Description

Syntax errors thrown from user actions.

Definition at line 548 of file GenesysParser.h.

7.172.2 Constructor & Destructor Documentation

7.172.2.1 syntax_error() [1/2]

```
yy::genesyspp_parser::syntax_error::syntax_error (
    const location_type & l,
    const std::string & m ) [inline]
```

Definition at line 550 of file GenesysParser.h.

7.172.2.2 syntax_error() [2/2]

```
yy::genesyspp_parser::syntax_error::syntax_error (
    const syntax_error & s ) [inline]
```

Definition at line 555 of file GenesysParser.h.

7.172.2.3 ~syntax_error()

```
yy::genesyspp_parser::syntax_error::~syntax_error ( )
```

Definition at line 158 of file GenesysParser.cpp.

7.172.3 Member Data Documentation

7.172.3.1 location

```
location_type yy::genesyspp_parser::syntax_error::location
```

Definition at line 562 of file GenesysParser.h.

The documentation for this struct was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.cpp

7.173 HypothesisTester_if::TestResult Class Reference

```
#include <HypothesisTester_if.h>
```

Collaboration diagram for HypothesisTester_if::TestResult:

Public Member Functions

- TestResult (double errorTypeI, double pvalue, bool rejectH0, double rejectionThreshold, double testStat)
- bool rejectH0 ()
- double pValue ()
- double errorTypeI ()
- double rejectionThreshold ()
- double testStat ()

7.173.1 Detailed Description

Definition at line 51 of file HypothesisTester_if.h.

7.173.2 Constructor & Destructor Documentation

7.173.2.1 TestResult()

```
HypothesisTester_if::TestResult::TestResult (
    double errorTypeI,
    double pvalue,
    bool rejectH0,
    double rejectionThreshold,
    double testStat ) [inline]
```

Definition at line 53 of file HypothesisTester_if.h.

Here is the call graph for this function:

7.173.3 Member Function Documentation

7.173.3.1 errorTypeI()

```
double HypothesisTester_if::TestResult::errorTypeI ( ) [inline]
```

Definition at line 66 of file HypothesisTester_if.h.

Here is the caller graph for this function:

7.173.3.2 pValue() `double HypothesisTester_if::TestResult::pValue ( ) [inline]`

Definition at line 63 of file HypothesisTester_if.h.

Here is the caller graph for this function:

7.173.3.3 rejectH0() `bool HypothesisTester_if::TestResult::rejectH0 ( ) [inline]`

Definition at line 60 of file HypothesisTester_if.h.

Here is the caller graph for this function:

7.173.3.4 rejectionThreshold() `double HypothesisTester_if::TestResult::rejectionThreshold ( ) [inline]`

Definition at line 69 of file HypothesisTester_if.h.

Here is the caller graph for this function:

7.173.3.5 testStat() `double HypothesisTester_if::TestResult::testStat ( ) [inline]`

Definition at line 72 of file HypothesisTester_if.h.

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/tools/HypothesisTester_if.h

7.174 yy::genesyspp_parser::token Struct Reference

Token kinds.

`#include <GenesysParser.h>`

Collaboration diagram for yy::genesyspp_parser::token:

Public Types

- enum token_kind_type {
  YYEMPTY = -2 , END = 0 , YYerror = 256 , YYUNDEF = 257 ,
  NUMD = 258 , NUMH = 259 , CTEZERO = 260 , oLE = 261 ,
  oGE = 262 , oEQ = 263 , oNE = 264 , oAND = 265 ,
  oOR = 266 , oNAND = 267 , oXOR = 268 , oNOT = 269 ,
  fSIN = 270 , fCOS = 271 , fROUND = 272 , fMOD = 273 ,
  fTRUNC = 274 , fFRAC = 275 , fEXP = 276 , fSQRT = 277 ,
  fLOG = 278 , fLN = 279 , fVAL = 280 , fEVAL = 281 ,
  fLENG = 282 , fRND1 = 283 , fEXPO = 284 , fNORM = 285 ,
  fUNIF = 286 , fWEIB = 287 , fLOGN = 288 , fGAMM = 289 ,

fERLA = 290 , fTRIA = 291 , fBETA = 292 , fDISC = 293 ,
fTNOW = 294 , fTFIN = 295 , fMAXREP = 296 , fNUMREP = 297 ,
fIDENT = 298 , cIF = 299 , cELSE = 300 , cFOR = 301 ,
cTO = 302 , cDO = 303 , ATRIB = 304 , CSTAT = 305 ,
fTAVG = 306 , ILLEGAL = 307 , RESOURCE = 308 , fNR = 309 ,
fMR = 310 , fIRF = 311 , fRESSEIZES = 312 , fSTATE = 313 ,
fSETSUM = 314 , fRESUTIL = 315 , QUEUE = 316 , fNQ = 317 ,
fFIRSTINQ = 318 , fLASTINQ = 319 , fSAQUE = 320 , fAQUE = 321 ,
fENTATRANK = 322 , SET = 323 , fNUMSET = 324 , VARI = 325 ,
FORM = 326 , fNUMGR = 327 , fATRGR = 328 , LPAREN = 329 ,
RPAREN = 330 , LBRACKET = 331 , RBRACKET = 332 , PLUS = 333 ,
MINUS = 334 , STAR = 335 , POWER = 336 , SLASH = 337 ,
LESS = 338 , GREATER = 339 , ASSIGN = 340 , COMMA = 341 ,
NEG = 342 }

- typedef token_kind_type yytokentype

  Backward compatibility alias (Bison 3.6).

7.174.1 Detailed Description

Token kinds.

Definition at line 566 of file GenesysParser.h.

7.174.2 Member Typedef Documentation

7.174.2.1 yytokentype `typedef token_kind_type yy::genesyspp_parser::token::yytokentype`

Backward compatibility alias (Bison 3.6).

Definition at line 661 of file GenesysParser.h.

7.174.3 Member Enumeration Documentation

7.174.3.1 token_kind_type `enum yy::genesyspp_parser::token::token_kind_type`

Enumerator

YYEMPTY		
END		
YYerror		
YYUNDEF		
NUMD		
NUMH		
CTEZERO		
oLE		
oGE		

Enumerator

oEQ		
oNE		
oAND		
oOR		
oNAND		
oXOR		
oNOT		
fSIN		
fCOS		
fROUND		
fMOD		
fTRUNC		
fFRAC		
fEXP		
fSQRT		
fLOG		
fLN		
fVAL		
fEVAL		
fLENG		
fRND1		
fEXPO		
fNORM		
fUNIF		
fWEIB		
fLOGN		
fGAMM		
fERLA		
fTRIA		
fBETA		
fDISC		
fTNOW		
fTFIN		
fMAXREP		
fNUMREP		
fIDENT		
cIF		
cELSE		
cFOR		
cTO		
cDO		
ATRIB		
CSTAT		
fTAVG		
ILLEGAL		
RESOURCE		
fNR		
fMR		
fIRF		
fRESSEIZES		

Enumerator

fSTATE		
fSETSUM		
fRESUTIL		
QUEUE		
fNQ		
fFIRSTINQ		
fLASTINQ		
fSAQUE		
fAQUE		
fENTATRANK		
SET		
fNUMSET		
VARI		
FORM		
fNUMGR		
fATRGR		
LPAREN		
RPAREN		
LBRACKET		
RBRACKET		
PLUS		
MINUS		
STAR		
POWER		
SLASH		
LESS		
GREATER		
ASSIGN		
COMMA		
NEG		

Definition at line 568 of file GenesysParser.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.h

7.175 TraceErrorEvent Class Reference

```
#include <TraceManager.h>
```

Inheritance diagram for TraceErrorEvent:

Collaboration diagram for TraceErrorEvent:

Public Member Functions

- TraceErrorEvent (std::string text, std::exception e)

- std::exception getException () const

7.175.1 Detailed Description

Definition at line 46 of file TraceManager.h.

7.175.2 Constructor & Destructor Documentation

7.175.2.1 TraceErrorEvent()

```
TraceErrorEvent::TraceErrorEvent (
    std::string text,
    std::exception e ) [inline]
```

Definition at line 48 of file TraceManager.h.

7.175.3 Member Function Documentation

7.175.3.1 getException()

```
std::exception TraceErrorEvent::getException ( ) const [inline]
```

Definition at line 51 of file TraceManager.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/TraceManager.h

7.176 TraceEvent Class Reference

```
#include <TraceManager.h>
```

Inheritance diagram for TraceEvent:

Collaboration diagram for TraceEvent:

Public Member Functions

- TraceEvent (Util::TraceLevel level, std::string text)
- TraceEvent (std::string text, Util::TraceLevel level=Util::TraceLevel::L8_detailed)
- Util::TraceLevel getTracelevel () const
- std::string getText () const

7.176.1 Detailed Description

Definition at line 25 of file TraceManager.h.

7.176.2 Constructor & Destructor Documentation

7.176.2.1 TraceEvent() [1/2]

```
TraceEvent::TraceEvent (
    Util::TraceLevel level,
    std::string text ) [inline]
```

Definition at line 27 of file TraceManager.h.

7.176.2.2 TraceEvent() [2/2]

```
TraceEvent::TraceEvent (
    std::string text,
    Util::TraceLevel level = Util::TraceLevel::L8_detailed ) [inline]
```

Definition at line 31 of file TraceManager.h.

7.176.3 Member Function Documentation

7.176.3.1 getText() `std::string TraceEvent::getText ( ) const [inline]`

Definition at line 38 of file TraceManager.h.

Here is the caller graph for this function:

7.176.3.2 getTracelevel() `Util::TraceLevel TraceEvent::getTracelevel ( ) const [inline]`

Definition at line 35 of file TraceManager.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/TraceManager.h

7.177 TraceManager Class Reference

```
#include <TraceManager.h>
```

Collaboration diagram for TraceManager:

Public Member Functions

- TraceManager (Simulator *simulator)
- virtual ~TraceManager ()=default
- void addTraceHandler (traceListener traceListener)
- void addTraceReportHandler (traceListener traceReportListener)
- void addTraceSimulationHandler (traceSimulationListener traceSimulationListener)
- void addTraceErrorHandler (traceErrorListener traceErrorListener)
- template<typename Class >
  void addTraceHandler (Class *object, void(Class::*function)(TraceEvent))
- template<typename Class >
  void addTraceErrorHandler (Class *object, void(Class::*function)(TraceErrorEvent))
- template<typename Class >
  void addTraceReportHandler (Class *object, void(Class::*function)(TraceEvent))
- template<typename Class >
  void addTraceSimulationHandler (Class *object, void(Class::*function)(TraceSimulationEvent))
- void trace (Util::TraceLevel level, std::string text)
- void traceError (std::exception e, std::string text)
- void traceReport (Util::TraceLevel level, std::string text)
- void traceSimulation (Util::TraceLevel level, double time, Entity *entity, ModelComponent *component, std↩::string text)
- void trace (std::string text, Util::TraceLevel level=Util::TraceLevel::L8_detailed)
- void traceError (std::string text, std::exception e)
- void traceReport (std::string text, Util::TraceLevel level=Util::TraceLevel::L2_results)
- void traceSimulation (double time, Entity *entity, ModelComponent *component, std::string text, Util::TraceLevel level=Util::TraceLevel::L8_detailed)
- List< std::string > * errorMessages () const
- void setTraceLevel (Util::TraceLevel _traceLevel)
- Util::TraceLevel getTraceLevel () const
- Simulator * getParentSimulator () const

7.177.1 Detailed Description

The TraceManager is used to trace back model simulation information and track/debug the simulation. It works as the model simulation output (cout) and allows external methods to hook up such output as listeners.

Definition at line 106 of file TraceManager.h.

7.177.2 Constructor & Destructor Documentation

7.177.2.1 TraceManager()

```
TraceManager::TraceManager (
    Simulator * simulator )
```

Definition at line 19 of file TraceManager.cpp.

7.177.2.2 ~TraceManager()

```
virtual TraceManager::~TraceManager ( ) [virtual], [default]
```

7.177.3 Member Function Documentation

7.177.3.1 addTraceErrorHandler() [1/2]

```
template<typename Class >
void TraceManager::addTraceErrorHandler (
    Class * object,
    void(Class::*)(TraceErrorEvent) function )
```

Definition at line 166 of file TraceManager.h.

Here is the call graph for this function:

7.177.3.2 addTraceErrorHandler() [2/2]

```
void TraceManager::addTraceErrorHandler (
            traceErrorListener traceErrorListener )
```

Definition at line 44 of file TraceManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.177.3.3 addTraceHandler() [1/2]

```
template<typename Class >
void TraceManager::addTraceHandler (
            Class * object,
            void(Class::*)(TraceEvent) function )
```

Definition at line 163 of file TraceManager.h.

Here is the call graph for this function:

7.177.3.4 addTraceHandler() [2/2]

```
void TraceManager::addTraceHandler (
    traceListener traceListener )
```

Definition at line 36 of file TraceManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.177.3.5 addTraceReportHandler() [1/2]

```
template<typename Class >
void TraceManager::addTraceReportHandler (
            Class * object,
            void(Class::*)(TraceEvent) function )
```

Definition at line 169 of file TraceManager.h.

Here is the call graph for this function:

7.177.3.6 addTraceReportHandler() [2/2]

```
void TraceManager::addTraceReportHandler (
        traceListener traceReportListener )
```

Definition at line 48 of file TraceManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.177.3.7 addTraceSimulationHandler() [1/2]

```
template<typename Class >
void TraceManager::addTraceSimulationHandler (
            Class * object,
            void(Class::*)(TraceSimulationEvent) function )
```

Definition at line 172 of file TraceManager.h.

Here is the call graph for this function:

7.177.3.8 addTraceSimulationHandler() [2/2]

```
void TraceManager::addTraceSimulationHandler (
    traceSimulationListener traceSimulationListener )
```

Definition at line 40 of file TraceManager.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.177.3.9 errorMessages() `List< std::string > * TraceManager::errorMessages ( ) const`

Definition at line 122 of file TraceManager.cpp.

7.177.3.10 getParentSimulator() `Simulator * TraceManager::getParentSimulator ( ) const`

Definition at line 32 of file TraceManager.cpp.

7.177.3.11 getTraceLevel() `Util::TraceLevel TraceManager::getTraceLevel ( ) const`

Definition at line 28 of file TraceManager.cpp.

7.177.3.12 setTraceLevel() `void TraceManager::setTraceLevel ( Util::TraceLevel _traceLevel )`

Definition at line 24 of file TraceManager.cpp.

Here is the caller graph for this function:

7.177.3.13 trace() [1/2]

```
void TraceManager::trace (
    std::string text,
    Util::TraceLevel level = Util::TraceLevel::L8_detailed )
```

Definition at line 56 of file TraceManager.cpp.

Here is the call graph for this function:

7.177.3.14 trace() [2/2]

```
void TraceManager::trace (
    Util::TraceLevel level,
    std::string text )
```

Definition at line 52 of file TraceManager.cpp.

7.177.3.15 traceError() [1/2]

```
void TraceManager::traceError (
    std::exception e,
    std::string text )
```

Definition at line 72 of file TraceManager.cpp.

Here is the caller graph for this function:

7.177.3.16 traceError() [2/2]

```
void TraceManager::traceError (
    std::string text,
    std::exception e )
```

Definition at line 76 of file TraceManager.cpp.

Here is the call graph for this function:

7.177.3.17 traceReport() [1/2]

```
void TraceManager::traceReport (
    std::string text,
    Util::TraceLevel level = Util::TraceLevel::L2_results )
```

Definition at line 109 of file TraceManager.cpp.

Here is the call graph for this function:

7.177.3.18 traceReport() [2/2]

```
void TraceManager::traceReport (
    Util::TraceLevel level,
    std::string text )
```

Definition at line 105 of file TraceManager.cpp.

Here is the caller graph for this function:

7.177.3.19 traceSimulation() [1/2]

```
void TraceManager::traceSimulation (
    double time,
    Entity * entity,
    ModelComponent * component,
    std::string text,
    Util::TraceLevel level = Util::TraceLevel::L8_detailed )
```

Definition at line 92 of file TraceManager.cpp.

Here is the call graph for this function:

7.177.3.20 traceSimulation() [2/2]

```
void TraceManager::traceSimulation (
    Util::TraceLevel level,
    double time,
    Entity * entity,
    ModelComponent * component,
    std::string text )
```

Definition at line 88 of file TraceManager.cpp.

Here is the caller graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/TraceManager.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/TraceManager.cpp

7.178 TraceSimulationEvent Class Reference

```
#include <TraceManager.h>
```

Inheritance diagram for TraceSimulationEvent:

Collaboration diagram for TraceSimulationEvent:

Public Member Functions

- TraceSimulationEvent (Util::TraceLevel level, double time, Entity *entity, ModelComponent *component, std::string text)
- ModelComponent * getComponent () const
- Entity * getEntity () const
- double getTime () const

7.178.1 Detailed Description

Definition at line 58 of file TraceManager.h.

7.178.2 Constructor & Destructor Documentation

7.178.2.1 TraceSimulationEvent()

```
TraceSimulationEvent::TraceSimulationEvent (
    Util::TraceLevel level,
    double time,
    Entity * entity,
    ModelComponent * component,
    std::string text ) [inline]
```

Definition at line 60 of file TraceManager.h.

7.178.3 Member Function Documentation

7.178.3.1 getComponent()

```
ModelComponent* TraceSimulationEvent::getComponent ( ) const [inline]
```

Definition at line 65 of file TraceManager.h.

7.178.3.2 getEntity()

```
Entity* TraceSimulationEvent::getEntity ( ) const [inline]
```

Definition at line 68 of file TraceManager.h.

7.178.3.3 getTime()

```
double TraceSimulationEvent::getTime ( ) const [inline]
```

Definition at line 71 of file TraceManager.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/TraceManager.h

7.179 TraceSimulationProcess Class Reference

```
#include <TraceManager.h>
```

Inheritance diagram for TraceSimulationProcess:

Collaboration diagram for TraceSimulationProcess:

Public Member Functions

- TraceSimulationProcess (std::string text, Util::TraceLevel level=Util::TraceLevel::L8_detailed)

7.179.1 Detailed Description

Events related to simulation "process" (usually process analyser), associated to entire replication or simulation events (begin/end/pause of replication/simulation)

Todo : CLASS NOT FULLY IMPLEMENTED (to be implemented for process analyser)

Definition at line 85 of file TraceManager.h.

7.179.2 Constructor & Destructor Documentation

7.179.2.1 TraceSimulationProcess()

```
TraceSimulationProcess::TraceSimulationProcess (
    std::string text,
    Util::TraceLevel level = Util::TraceLevel::L8_detailed ) [inline]
```

Definition at line 87 of file TraceManager.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/simulator/TraceManager.h

7.180 Traits< T > Struct Template Reference

```
#include <Traits.h>
```

Collaboration diagram for Traits< T >:

7.180.1 Detailed Description

template<typename T>
struct Traits< T >

Definition at line 52 of file Traits.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/Traits.h

7.181 Traits< GenesysApplication_if > Struct Reference

```
#include <Traits.h>
```

Collaboration diagram for Traits< GenesysApplication_if >:

Public Types

- typedef Smart_OnEvent Application

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L6_arrival

7.181.1 Detailed Description

Configure the Genesys Application to be compiled and executed

Definition at line 58 of file Traits.h.

7.181.2 Member Typedef Documentation

7.181.2.1 Application `typedef Smart_OnEvent Traits< GenesysApplication_if >::Application`

Definition at line 67 of file Traits.h.

7.181.3 Member Data Documentation

7.181.3.1 traceLevel `const Util::TraceLevel Traits< GenesysApplication_if >::traceLevel = Util::TraceLevel::L6_arrival [static]`

Definition at line 59 of file Traits.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/Traits.h

7.182 Traits< HypothesisTester_if > Struct Reference

`#include <Traits.h>`

Collaboration diagram for Traits< HypothesisTester_if >:

Public Types

- typedef HypothesisTesterDefaultImpl1 Implementation

Static Public Attributes

- static constexpr unsigned int ConfidenceLevel = 95

7.182.1 Detailed Description

Configure the Hypothesis Tester to be used

Definition at line 100 of file Traits.h.

7.182.2 Member Typedef Documentation

7.182.2.1 Implementation `typedef HypothesisTesterDefaultImpl1 Traits< HypothesisTester_if >↩ ::Implementation`

Definition at line 101 of file Traits.h.

7.182.3 Member Data Documentation

7.182.3.1 ConfidenceLevel `constexpr unsigned int Traits< HypothesisTester_if >::Confidence↩ Level = 95 [static], [constexpr]`

Definition at line 102 of file Traits.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/Traits.h

7.183 Traits< Solver_if > Struct Reference

`#include <Traits.h>`

Collaboration diagram for Traits< Solver_if >:

Public Types

- typedef SolverDefaultImpl1 Implementation

Static Public Attributes

- static constexpr double Precision = 1e-5
- static constexpr unsigned int MaxSteps = 1e2

7.183.1 Detailed Description

Configure the Solver to be used

Definition at line 91 of file Traits.h.

7.183.2 Member Typedef Documentation

7.183.2.1 Implementation `typedef SolverDefaultImpl1 Traits< Solver_if >::Implementation`

Definition at line 92 of file Traits.h.

7.183.3 Member Data Documentation

7.183.3.1 MaxSteps `constexpr unsigned int Traits< Solver_if >::MaxSteps = 1e2 [static], [constexpr]`

Definition at line 94 of file Traits.h.

7.183.3.2 Precision `constexpr double Traits< Solver_if >::Precision = 1e-5 [static], [constexpr]`

Definition at line 93 of file Traits.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/Traits.h

7.184 TraitsKernel< T > Struct Template Reference

```
#include <TraitsKernel.h>
```

Collaboration diagram for TraitsKernel< T >:

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results

7.184.1 Detailed Description

template<typename T>
struct TraitsKernel< T >

Definition at line 37 of file TraitsKernel.h.

7.184.2 Member Data Documentation

7.184.2.1 traceLevel `template<typename T >`

`const Util::TraceLevel TraitsKernel< T >::traceLevel = Util::TraceLevel::L2_results [static]`

Definition at line 38 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.185 TraitsKernel< Collector_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< Collector_if >:

Public Types

- typedef CollectorDatafileDefaultImpl1 Implementation
- typedef double DataType

7.185.1 Detailed Description

Definition at line 111 of file TraitsKernel.h.

7.185.2 Member Typedef Documentation

7.185.2.1 DataType `typedef double TraitsKernel< Collector_if >::DataType`

Definition at line 113 of file TraitsKernel.h.

7.185.2.2 Implementation `typedef CollectorDatafileDefaultImpl1 TraitsKernel< Collector_if >↩ ::Implementation`

Definition at line 112 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.186 TraitsKernel< Model > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< Model >:

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L5_event

7.186.1 Detailed Description

Definition at line 64 of file TraitsKernel.h.

7.186.2 Member Data Documentation

7.186.2.1 traceLevel `const Util::TraceLevel TraitsKernel< Model >::traceLevel = Util::TraceLevel::L5_event [static]`

Definition at line 65 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.187 TraitsKernel< ModelChecker_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< ModelChecker_if >:

Public Types

- typedef ModelCheckerDefaultImpl1 Implementation

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results

7.187.1 Detailed Description

Definition at line 80 of file TraitsKernel.h.

7.187.2 Member Typedef Documentation

7.187.2.1 Implementation `typedef ModelCheckerDefaultImpl1 TraitsKernel< ModelChecker_if >↩ ::Implementation`

Definition at line 81 of file TraitsKernel.h.

7.187.3 Member Data Documentation

7.187.3.1 traceLevel `const Util::TraceLevel TraitsKernel< ModelChecker_if >::traceLevel = Util::TraceLevel::L2_results [static]`

Definition at line 82 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.188 TraitsKernel< ModelComponent > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< ModelComponent >:

Public Types

- typedef StatisticsDefaultImpl1 StatisticsCollector_StatisticsImplementation
- typedef CollectorDefaultImpl1 StatisticsCollector_CollectorImplementation

Static Public Attributes

- static constexpr bool reportStatistics = true
- static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results

7.188.1 Detailed Description

Definition at line 68 of file TraitsKernel.h.

7.188.2 Member Typedef Documentation

7.188.2.1 StatisticsCollector_CollectorImplementation `typedef CollectorDefaultImpl1 TraitsKernel< ModelComponent >::StatisticsCollector_CollectorImplementation`

Definition at line 70 of file TraitsKernel.h.

7.188.2.2 StatisticsCollector_StatisticsImplementation `typedef StatisticsDefaultImpl1 TraitsKernel< ModelComponent >::StatisticsCollector_StatisticsImplementation`

Definition at line 69 of file TraitsKernel.h.

7.188.3 Member Data Documentation

7.188.3.1 reportStatistics `constexpr bool TraitsKernel< ModelComponent >::reportStatistics = true [static], [constexpr]`

Definition at line 71 of file TraitsKernel.h.

7.188.3.2 traceLevel `const Util::TraceLevel TraitsKernel< ModelComponent >::traceLevel = Util::TraceLevel::L2_ [static]`

Definition at line 72 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.189 TraitsKernel< ModelElement > Struct Reference

```
#include <TraitsKernel.h>
```

Collaboration diagram for TraitsKernel< ModelElement >:

Static Public Attributes

- static constexpr bool reportStatistics = true
- static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results

7.189.1 Detailed Description

Definition at line 75 of file TraitsKernel.h.

7.189.2 Member Data Documentation

7.189.2.1 reportStatistics `constexpr bool TraitsKernel< ModelElement >::reportStatistics = true [static], [constexpr]`

Definition at line 76 of file TraitsKernel.h.

7.189.2.2 traceLevel

`const Util::TraceLevel TraitsKernel< ModelElement >::traceLevel = Util::TraceLevel::L2_re`
`[static]`

Definition at line 77 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.190 TraitsKernel< ModelPersistence_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< ModelPersistence_if >:

Public Types

- typedef ModelPersistenceDefaultImpl1 Implementation

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results

7.190.1 Detailed Description

Definition at line 85 of file TraitsKernel.h.

7.190.2 Member Typedef Documentation

7.190.2.1 Implementation `typedef ModelPersistenceDefaultImpl1 TraitsKernel< ModelPersistence_if >::Implementation`

Definition at line 86 of file TraitsKernel.h.

7.190.3 Member Data Documentation

7.190.3.1 traceLevel `const Util::TraceLevel TraitsKernel< ModelPersistence_if >::traceLevel = Util::TraceLevel::L2_results [static]`

Definition at line 87 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.191 TraitsKernel< Parser_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< Parser_if >:

Public Types

- typedef ParserDefaultImpl2 Implementation

7.191.1 Detailed Description

Definition at line 56 of file TraitsKernel.h.

7.191.2 Member Typedef Documentation

7.191.2.1 Implementation `typedef ParserDefaultImpl2 TraitsKernel< Parser_if >::Implementation`

Definition at line 57 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.192 TraitsKernel< PluginConnector_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< PluginConnector_if >:

Public Types

- typedef PluginConnectorDummyImpl1 Implementation

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L4_warning

7.192.1 Detailed Description

Definition at line 51 of file TraitsKernel.h.

7.192.2 Member Typedef Documentation

7.192.2.1 Implementation `typedef PluginConnectorDummyImpl1 TraitsKernel< PluginConnector_if >::Implementation`

Definition at line 52 of file TraitsKernel.h.

7.192.3 Member Data Documentation

7.192.3.1 traceLevel `const Util::TraceLevel TraitsKernel< PluginConnector_if >::traceLevel = Util::TraceLevel::L4_warning [static]`

Definition at line 53 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.193 TraitsKernel< Sampler_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< Sampler_if >:

Public Types

- typedef SamplerDefaultImpl1 Implementation
- typedef SamplerDefaultImpl1::DefaultImpl1RNG_Parameters Parameters

7.193.1 Detailed Description

Definition at line 106 of file TraitsKernel.h.

7.193.2 Member Typedef Documentation

7.193.2.1 Implementation `typedef SamplerDefaultImpl1 TraitsKernel< Sampler_if >::Implementation`

Definition at line 107 of file TraitsKernel.h.

7.193.2.2 Parameters `typedef SamplerDefaultImpl1::DefaultImpl1RNG_Parameters TraitsKernel< Sampler_if >::Parameters`

Definition at line 108 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.194 TraitsKernel< SimulationReporter_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< SimulationReporter_if >:

Public Types

- typedef SimulationReporterDefaultImpl1 Implementation
- typedef Counter CounterImplementation

Static Public Attributes

- static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results

7.194.1 Detailed Description

Definition at line 45 of file TraitsKernel.h.

7.194.2 Member Typedef Documentation

7.194.2.1 CounterImplementation `typedef Counter TraitsKernel< SimulationReporter_if >::CounterImplementation`

Definition at line 47 of file TraitsKernel.h.

7.194.2.2 Implementation `typedef SimulationReporterDefaultImpl1 TraitsKernel< SimulationReporter_if >::Implementation`

Definition at line 46 of file TraitsKernel.h.

7.194.3 Member Data Documentation

7.194.3.1 traceLevel `const Util::TraceLevel TraitsKernel< SimulationReporter_if >::traceLevel = Util::TraceLevel::L2_results [static]`

Definition at line 48 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.195 TraitsKernel< Statistics_if > Struct Reference

`#include <TraitsKernel.h>`

Collaboration diagram for TraitsKernel< Statistics_if >:

Public Types

- typedef StatisticsDefaultImpl1 Implementation
- typedef CollectorDefaultImpl1 CollectorImplementation
- typedef double DataType

Static Public Attributes

- static constexpr double SignificanceLevel = 0.05

7.195.1 Detailed Description

Definition at line 93 of file TraitsKernel.h.

7.195.2 Member Typedef Documentation

7.195.2.1 CollectorImplementation `typedef CollectorDefaultImpl1 TraitsKernel< Statistics_if >::CollectorImplementation`

Definition at line 95 of file TraitsKernel.h.

7.195.2.2 DataType `typedef double TraitsKernel< Statistics_if >::DataType`

Definition at line 96 of file TraitsKernel.h.

7.195.2.3 Implementation `typedef StatisticsDefaultImpl1 TraitsKernel< Statistics_if >::Implementation`

Definition at line 94 of file TraitsKernel.h.

7.195.3 Member Data Documentation

7.195.3.1 SignificanceLevel `constexpr double TraitsKernel< Statistics_if >::SignificanceLevel = 0.05 [static], [constexpr]`

Definition at line 97 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/TraitsKernel.h

7.196 TraitsKernel< StatisticsDatafile_if > Struct Reference

```
#include <TraitsKernel.h>
```

Collaboration diagram for TraitsKernel< StatisticsDatafile_if >:

Public Types

- typedef StatisticsDatafileDefaultImpl1 Implementation
- typedef CollectorDatafileDefaultImpl1 CollectorImplementation

Static Public Attributes

- static constexpr double SignificanceLevel = 0.05

7.196.1 Detailed Description

Definition at line 100 of file TraitsKernel.h.

7.196.2 Member Typedef Documentation

7.196.2.1 CollectorImplementation `typedef CollectorDatafileDefaultImpl1 TraitsKernel< StatisticsDatafile_if >::CollectorImplementation`

Definition at line 102 of file TraitsKernel.h.

7.196.2.2 Implementation `typedef StatisticsDatafileDefaultImpl1 TraitsKernel< StatisticsDatafile_if >::Implementation`

Definition at line 101 of file TraitsKernel.h.

7.196.3 Member Data Documentation

7.196.3.1 SignificanceLevel `constexpr double TraitsKernel< StatisticsDatafile_if >::SignificanceLevel = 0.05 [static], [constexpr]`

Definition at line 103 of file TraitsKernel.h.

The documentation for this struct was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/TraitsKernel.h

7.197 Ui_DialogAbout Class Reference

```
#include <ui_dialogabout.h>
```

Inheritance diagram for Ui_DialogAbout:

Collaboration diagram for Ui_DialogAbout:

Public Member Functions

- void setupUi (QDialog ∗DialogAbout)
- void retranslateUi (QDialog ∗DialogAbout)

Public Attributes

- QVBoxLayout * verticalLayout_3
- QHBoxLayout * horizontalLayout_2
- QLabel * labelIcon
- QVBoxLayout * verticalLayout_2
- QLabel * label_2
- QLabel * labelAuthor
- QLabel * labelInstitution
- QLabel * labelVersionGenesys
- QTextEdit * textEditDescription
- QDialogButtonBox * buttonOk

7.197.1 Detailed Description

Definition at line 24 of file ui_dialogabout.h.

7.197.2 Member Function Documentation

7.197.2.1 retranslateUi()

```
void Ui_DialogAbout::retranslateUi (
    QDialog * DialogAbout ) [inline]
```

Definition at line 116 of file ui_dialogabout.h.

Here is the caller graph for this function:

7.197.2.2 setupUi()

```
void Ui_DialogAbout::setupUi (
    QDialog * DialogAbout ) [inline]
```

Definition at line 38 of file ui_dialogabout.h.

Here is the call graph for this function:

7.197.3 Member Data Documentation

7.197.3.1 buttonOk `QDialogButtonBox* Ui_DialogAbout::buttonOk`

Definition at line 36 of file ui_dialogabout.h.

7.197.3.2 horizontalLayout_2 `QHBoxLayout* Ui_DialogAbout::horizontalLayout_2`

Definition at line 28 of file ui_dialogabout.h.

7.197.3.3 label_2 `QLabel* Ui_DialogAbout::label_2`

Definition at line 31 of file ui_dialogabout.h.

7.197.3.4 labelAuthor `QLabel* Ui_DialogAbout::labelAuthor`

Definition at line 32 of file ui_dialogabout.h.

7.197.3.5 labelIcon `QLabel* Ui_DialogAbout::labelIcon`

Definition at line 29 of file ui_dialogabout.h.

7.197.3.6 labelInstitution `QLabel* Ui_DialogAbout::labelInstitution`

Definition at line 33 of file ui_dialogabout.h.

7.197.3.7 labelVersionGenesys `QLabel* Ui_DialogAbout::labelVersionGenesys`

Definition at line 34 of file ui_dialogabout.h.

7.197.3.8 textEditDescription `QTextEdit* Ui_DialogAbout::textEditDescription`

Definition at line 35 of file ui_dialogabout.h.

7.197.3.9 verticalLayout_2 `QVBoxLayout* Ui_DialogAbout::verticalLayout_2`

Definition at line 30 of file ui_dialogabout.h.

7.197.3.10 verticalLayout_3 `QVBoxLayout* Ui_DialogAbout::verticalLayout_3`

Definition at line 27 of file ui_dialogabout.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/ui_dialogabout.h

7.198 Ui_DialogModelInformation Class Reference

```
#include <ui_dialogmodelinformation.h>
```

Inheritance diagram for Ui_DialogModelInformation:

Collaboration diagram for Ui_DialogModelInformation:

Public Member Functions

- void setupUi (QDialog ∗DialogModelInformation)
- void retranslateUi (QDialog ∗DialogModelInformation)

Public Attributes

- QVBoxLayout ∗ verticalLayout
- QHBoxLayout ∗ horizontalLayout_2
- QLabel ∗ label_2
- QLineEdit ∗ lineEdit_AnalystName
- QHBoxLayout ∗ horizontalLayout_3
- QLabel ∗ label_4
- QLineEdit ∗ lineEdit_ProjectTitle
- QHBoxLayout ∗ horizontalLayout
- QLabel ∗ label
- QLineEdit ∗ lineEdit_ModelName
- QHBoxLayout ∗ horizontalLayout_4
- QLabel ∗ label_3
- QLineEdit ∗ lineEdit_Description
- QHBoxLayout ∗ horizontalLayout_5
- QLabel ∗ label_5
- QLineEdit ∗ lineEdit_Version
- QDialogButtonBox ∗ buttonBox

7.198.1 Detailed Description

Definition at line 24 of file ui_dialogmodelinformation.h.

7.198.2 Member Function Documentation

7.198.2.1 retranslateUi()

```
void Ui_DialogModelInformation::retranslateUi (
    QDialog * DialogModelInformation ) [inline]
```

Definition at line 154 of file ui_dialogmodelinformation.h.

Here is the caller graph for this function:

7.198.2.2 setupUi()

```
void Ui_DialogModelInformation::setupUi (
    QDialog * DialogModelInformation ) [inline]
```

Definition at line 45 of file ui_dialogmodelinformation.h.

Here is the call graph for this function:

7.198.3 Member Data Documentation

7.198.3.1 buttonBox

```
QDialogButtonBox* Ui_DialogModelInformation::buttonBox
```

Definition at line 43 of file ui_dialogmodelinformation.h.

7.198.3.2 horizontalLayout

```
QHBoxLayout* Ui_DialogModelInformation::horizontalLayout
```

Definition at line 34 of file ui_dialogmodelinformation.h.

7.198.3.3 horizontalLayout_2 `QHBoxLayout* Ui_DialogModelInformation::horizontalLayout_2`

Definition at line 28 of file ui_dialogmodelinformation.h.

7.198.3.4 horizontalLayout_3 `QHBoxLayout* Ui_DialogModelInformation::horizontalLayout_3`

Definition at line 31 of file ui_dialogmodelinformation.h.

7.198.3.5 horizontalLayout_4 `QHBoxLayout* Ui_DialogModelInformation::horizontalLayout_4`

Definition at line 37 of file ui_dialogmodelinformation.h.

7.198.3.6 horizontalLayout_5 `QHBoxLayout* Ui_DialogModelInformation::horizontalLayout_5`

Definition at line 40 of file ui_dialogmodelinformation.h.

7.198.3.7 label `QLabel* Ui_DialogModelInformation::label`

Definition at line 35 of file ui_dialogmodelinformation.h.

7.198.3.8 label_2 `QLabel* Ui_DialogModelInformation::label_2`

Definition at line 29 of file ui_dialogmodelinformation.h.

7.198.3.9 label_3 `QLabel* Ui_DialogModelInformation::label_3`

Definition at line 38 of file ui_dialogmodelinformation.h.

7.198.3.10 label_4 `QLabel* Ui_DialogModelInformation::label_4`

Definition at line 32 of file ui_dialogmodelinformation.h.

7.198.3.11 label_5 `QLabel* Ui_DialogModelInformation::label_5`

Definition at line 41 of file ui_dialogmodelinformation.h.

7.198.3.12 lineEdit_AnalystName `QLineEdit* Ui_DialogModelInformation::lineEdit_AnalystName`

Definition at line 30 of file ui_dialogmodelinformation.h.

7.198.3.13 lineEdit_Description `QLineEdit* Ui_DialogModelInformation::lineEdit_Description`

Definition at line 39 of file ui_dialogmodelinformation.h.

7.198.3.14 lineEdit_ModelName `QLineEdit* Ui_DialogModelInformation::lineEdit_ModelName`

Definition at line 36 of file ui_dialogmodelinformation.h.

7.198.3.15 lineEdit_ProjectTitle `QLineEdit* Ui_DialogModelInformation::lineEdit_ProjectTitle`

Definition at line 33 of file ui_dialogmodelinformation.h.

7.198.3.16 lineEdit_Version `QLineEdit* Ui_DialogModelInformation::lineEdit_Version`

Definition at line 42 of file ui_dialogmodelinformation.h.

7.198.3.17 verticalLayout `QVBoxLayout* Ui_DialogModelInformation::verticalLayout`

Definition at line 27 of file ui_dialogmodelinformation.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/ui_dialogmodelinformation.h

7.199 Ui_formPlugins Class Reference

```
#include <ui_formPlugins.h>
```

Inheritance diagram for Ui_formPlugins:

Collaboration diagram for Ui_formPlugins:

Public Member Functions

- void setupUi (QDialog ∗formPlugins)
- void retranslateUi (QDialog ∗formPlugins)

Public Attributes

- QDialogButtonBox * buttonBox

7.199.1 Detailed Description

Definition at line 19 of file ui_formPlugins.h.

7.199.2 Member Function Documentation

7.199.2.1 retranslateUi()

```
void Ui_formPlugins::retranslateUi (
    QDialog * formPlugins ) [inline]
```

Definition at line 42 of file ui_formPlugins.h.

Here is the caller graph for this function:

7.199.2.2 setupUi()

```
void Ui_formPlugins::setupUi (
    QDialog * formPlugins ) [inline]
```

Definition at line 24 of file ui_formPlugins.h.

Here is the call graph for this function:

7.199.3 Member Data Documentation

7.199.3.1 buttonBox `QDialogButtonBox* Ui_formPlugins::buttonBox`

Definition at line 22 of file ui_formPlugins.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/ui_formPlugins.h

7.200 Ui_MainWindow Class Reference

```
#include <ui_mainwindow.h>
```

Inheritance diagram for Ui_MainWindow:

Collaboration diagram for Ui_MainWindow:

Public Member Functions

- void setupUi (QMainWindow ∗MainWindow)
- void retranslateUi (QMainWindow ∗MainWindow)
- void setupUi (QMainWindow ∗MainWindow)
- void retranslateUi (QMainWindow ∗MainWindow)

Public Attributes

- QAction ∗ actionNew
- QAction ∗ actionOpen
- QAction ∗ actionClose
- QAction ∗ actionSave
- QAction ∗ actionSave_as
- QAction ∗ actionExit
- QAction ∗ actionUndo
- QAction ∗ actionRedo
- QAction ∗ actionCut
- QAction ∗ actionCopy
- QAction ∗ actionPaste
- QAction ∗ actionFind
- QAction ∗ actionReplace
- QAction ∗ actionGroup
- QAction ∗ actionZoom_In
- QAction ∗ actionZoom_Out
- QAction ∗ actionZoom_Factor
- QAction ∗ actionGrid
- QAction ∗ actionSnap
- QAction ∗ actionInput_Analyser
- QAction ∗ actionOutput_Analyser
- QAction ∗ actionScenario_Manager
- QAction ∗ actionOptimizaton
- QAction ∗ actionPreferences
- QAction ∗ actionCheck_Model
- QAction ∗ actionStart
- QAction ∗ actionStep
- QAction ∗ actionStop
- QAction ∗ actionRun_Control
- QAction ∗ actionComponents
- QAction ∗ actionElements
- QAction ∗ actionModels
- QAction ∗ actionConsole

- QAction * actionProperties
- QAction * actionEditor
- QAction * actionDocumentation
- QAction * actionSupport
- QAction * actionAbout
- QAction * actionTop
- QAction * actionBottom
- QAction * actionInformation
- QAction * actionReport
- QAction * actionComponents_2
- QAction * actionElements_2
- QAction * actionTrace
- QAction * actionEvents
- QAction * actionPause
- QAction * actionExit_2
- QAction * actionPreferences_2
- QAction * actionModels_2
- QAction * actionExperiments
- QAction * actionConnect
- QAction * actionZoom_In_2
- QAction * actionZoom_Out_2
- QAction * actionZoom_Facot
- QAction * actionGrid_2
- QAction * actionSnap_2
- QAction * actionArranje
- QAction * actionGroup_2
- QAction * actionPlugins_2
- QAction * actionInformation_2
- QAction * actionSimulation
- QAction * actionComponents_3
- QAction * actionElements_3
- QAction * actionLoad
- QAction * actionSave_2
- QAction * actionEvents_Diagram
- QAction * actionArranje_2
- QWidget * centralWidget
- QGridLayout * gridLayout
- QTabWidget * tabWidgetModels
- QWidget * tabModel1
- QHBoxLayout * horizontalLayout
- QGraphicsView * graphicsView
- QWidget * tabModel2
- QHBoxLayout * horizontalLayout_2
- QMenuBar * menuBar
- QMenu * menuEdit
- QMenu * menuModel
- QMenu * menuView_2
- QMenu * menuWindow
- QMenu * menuHelp
- QMenu * menuSimulator
- QStatusBar * statusBar
- QToolBar * toolBar
- QDockWidget * dockWidgetPlugins
- QWidget * dockWidgetPlugins_2
- QVBoxLayout * verticalLayout

- QListWidget * listWidget_Plugins
- QDockWidget * dockWidgetProperties
- QWidget * dockWidgetContents_Properties
- QVBoxLayout * verticalLayout_3
- QTableView * tableView_Properties
- QDockWidget * dockWidgetConsole
- QWidget * dockWidgetContents_Console
- QVBoxLayout * verticalLayout_4
- QTextEdit * textEdit_Console
- QDockWidget * dockWidgetReports
- QWidget * dockWidgetContents_Reports
- QVBoxLayout * verticalLayout_5
- QTextEdit * textEdit_Report
- QDockWidget * dockWidgetExperiments
- QWidget * dockWidgetContents_Experiments
- QListWidget * listWidget_Experiments

7.200.1 Detailed Description

Definition at line 35 of file ui_mainwindow.h.

7.200.2 Member Function Documentation

7.200.2.1 retranslateUi() [1/2]

```
void Ui_MainWindow::retranslateUi (
    QMainWindow * MainWindow ) [inline]
```

Definition at line 586 of file ui_mainwindow.h.

Here is the caller graph for this function:

7.200.2.2 retranslateUi() [2/2]

```
void Ui_MainWindow::retranslateUi (
    QMainWindow * MainWindow ) [inline]
```

Definition at line 590 of file ui_mainwindow.h.

7.200.2.3 setupUi() [1/2]

```
void Ui_MainWindow::setupUi (
    QMainWindow * MainWindow ) [inline]
```

Definition at line 141 of file ui_mainwindow.h.

Here is the call graph for this function:

Here is the caller graph for this function:

7.200.2.4 setupUi() [2/2]

```
void Ui_MainWindow::setupUi (
    QMainWindow * MainWindow ) [inline]
```

Definition at line 143 of file ui_mainwindow.h.

Here is the call graph for this function:

7.200.3 Member Data Documentation

7.200.3.1 actionAbout

```
QAction * Ui_MainWindow::actionAbout
```

Definition at line 74 of file ui_mainwindow.h.

7.200.3.2 actionArranje `QAction * Ui_MainWindow::actionArranje`

Definition at line 94 of file ui_mainwindow.h.

7.200.3.3 actionArranje_2 `QAction * Ui_MainWindow::actionArranje_2`

Definition at line 104 of file ui_mainwindow.h.

7.200.3.4 actionBottom `QAction * Ui_MainWindow::actionBottom`

Definition at line 76 of file ui_mainwindow.h.

7.200.3.5 actionCheck_Model `QAction * Ui_MainWindow::actionCheck_Model`

Definition at line 61 of file ui_mainwindow.h.

7.200.3.6 actionClose `QAction * Ui_MainWindow::actionClose`

Definition at line 39 of file ui_mainwindow.h.

7.200.3.7 actionComponents `QAction * Ui_MainWindow::actionComponents`

Definition at line 66 of file ui_mainwindow.h.

7.200.3.8 actionComponents_2 `QAction * Ui_MainWindow::actionComponents_2`

Definition at line 79 of file ui_mainwindow.h.

7.200.3.9 actionComponents_3 `QAction * Ui_MainWindow::actionComponents_3`

Definition at line 99 of file ui_mainwindow.h.

7.200.3.10 actionConnect `QAction * Ui_MainWindow::actionConnect`

Definition at line 88 of file ui_mainwindow.h.

7.200.3.11 actionConsole `QAction * Ui_MainWindow::actionConsole`

Definition at line 69 of file ui_mainwindow.h.

7.200.3.12 actionCopy `QAction * Ui_MainWindow::actionCopy`

Definition at line 46 of file ui_mainwindow.h.

7.200.3.13 actionCut `QAction * Ui_MainWindow::actionCut`

Definition at line 45 of file ui_mainwindow.h.

7.200.3.14 actionDocumentation `QAction * Ui_MainWindow::actionDocumentation`

Definition at line 72 of file ui_mainwindow.h.

7.200.3.15 actionEditor `QAction * Ui_MainWindow::actionEditor`

Definition at line 71 of file ui_mainwindow.h.

7.200.3.16 actionElements `QAction * Ui_MainWindow::actionElements`

Definition at line 67 of file ui_mainwindow.h.

7.200.3.17 actionElements_2 `QAction * Ui_MainWindow::actionElements_2`

Definition at line 80 of file ui_mainwindow.h.

7.200.3.18 actionElements_3 `QAction * Ui_MainWindow::actionElements_3`

Definition at line 100 of file ui_mainwindow.h.

7.200.3.19 actionEvents `QAction * Ui_MainWindow::actionEvents`

Definition at line 82 of file ui_mainwindow.h.

7.200.3.20 actionEvents_Diagram `QAction * Ui_MainWindow::actionEvents_Diagram`

Definition at line 103 of file ui_mainwindow.h.

7.200.3.21 actionExit `QAction * Ui_MainWindow::actionExit`

Definition at line 42 of file ui_mainwindow.h.

7.200.3.22 actionExit_2 `QAction * Ui_MainWindow::actionExit_2`

Definition at line 84 of file ui_mainwindow.h.

7.200.3.23 actionExperiments `QAction * Ui_MainWindow::actionExperiments`

Definition at line 87 of file ui_mainwindow.h.

7.200.3.24 actionFind `QAction * Ui_MainWindow::actionFind`

Definition at line 48 of file ui_mainwindow.h.

7.200.3.25 actionGrid `QAction * Ui_MainWindow::actionGrid`

Definition at line 54 of file ui_mainwindow.h.

7.200.3.26 actionGrid_2 `QAction * Ui_MainWindow::actionGrid_2`

Definition at line 92 of file ui_mainwindow.h.

7.200.3.27 actionGroup `QAction * Ui_MainWindow::actionGroup`

Definition at line 50 of file ui_mainwindow.h.

7.200.3.28 actionGroup_2 `QAction * Ui_MainWindow::actionGroup_2`

Definition at line 95 of file ui_mainwindow.h.

7.200.3.29 actionInformation `QAction * Ui_MainWindow::actionInformation`

Definition at line 77 of file ui_mainwindow.h.

7.200.3.30 actionInformation_2 `QAction * Ui_MainWindow::actionInformation_2`

Definition at line 97 of file ui_mainwindow.h.

7.200.3.31 actionInput_Analyser `QAction * Ui_MainWindow::actionInput_Analyser`

Definition at line 56 of file ui_mainwindow.h.

7.200.3.32 actionLoad `QAction * Ui_MainWindow::actionLoad`

Definition at line 101 of file ui_mainwindow.h.

7.200.3.33 actionModels `QAction * Ui_MainWindow::actionModels`

Definition at line 68 of file ui_mainwindow.h.

7.200.3.34 actionModels_2 `QAction * Ui_MainWindow::actionModels_2`

Definition at line 86 of file ui_mainwindow.h.

7.200.3.35 actionNew `QAction * Ui_MainWindow::actionNew`

Definition at line 37 of file ui_mainwindow.h.

7.200.3.36 actionOpen `QAction * Ui_MainWindow::actionOpen`

Definition at line 38 of file ui_mainwindow.h.

7.200.3.37 actionOptimizaton `QAction * Ui_MainWindow::actionOptimizaton`

Definition at line 59 of file ui_mainwindow.h.

7.200.3.38 actionOutput_Analyser `QAction * Ui_MainWindow::actionOutput_Analyser`

Definition at line 57 of file ui_mainwindow.h.

7.200.3.39 actionPaste `QAction * Ui_MainWindow::actionPaste`

Definition at line 47 of file ui_mainwindow.h.

7.200.3.40 actionPause `QAction * Ui_MainWindow::actionPause`

Definition at line 83 of file ui_mainwindow.h.

7.200.3.41 actionPlugins_2 `QAction * Ui_MainWindow::actionPlugins_2`

Definition at line 96 of file ui_mainwindow.h.

7.200.3.42 actionPreferences `QAction * Ui_MainWindow::actionPreferences`

Definition at line 60 of file ui_mainwindow.h.

7.200.3.43 actionPreferences_2 `QAction * Ui_MainWindow::actionPreferences_2`

Definition at line 85 of file ui_mainwindow.h.

7.200.3.44 actionProperties `QAction * Ui_MainWindow::actionProperties`

Definition at line 70 of file ui_mainwindow.h.

7.200.3.45 actionRedo `QAction * Ui_MainWindow::actionRedo`

Definition at line 44 of file ui_mainwindow.h.

7.200.3.46 actionReplace `QAction * Ui_MainWindow::actionReplace`

Definition at line 49 of file ui_mainwindow.h.

7.200.3.47 actionReport `QAction * Ui_MainWindow::actionReport`

Definition at line 78 of file ui_mainwindow.h.

7.200.3.48 actionRun_Control `QAction * Ui_MainWindow::actionRun_Control`

Definition at line 65 of file ui_mainwindow.h.

7.200.3.49 actionSave `QAction * Ui_MainWindow::actionSave`

Definition at line 40 of file ui_mainwindow.h.

7.200.3.50 actionSave_2 `QAction * Ui_MainWindow::actionSave_2`

Definition at line 102 of file ui_mainwindow.h.

7.200.3.51 actionSave_as `QAction * Ui_MainWindow::actionSave_as`

Definition at line 41 of file ui_mainwindow.h.

7.200.3.52 actionScenario_Manager `QAction * Ui_MainWindow::actionScenario_Manager`

Definition at line 58 of file ui_mainwindow.h.

7.200.3.53 actionSimulation `QAction * Ui_MainWindow::actionSimulation`

Definition at line 98 of file ui_mainwindow.h.

7.200.3.54 actionSnap `QAction * Ui_MainWindow::actionSnap`

Definition at line 55 of file ui_mainwindow.h.

7.200.3.55 actionSnap_2 `QAction * Ui_MainWindow::actionSnap_2`

Definition at line 93 of file ui_mainwindow.h.

7.200.3.56 actionStart `QAction * Ui_MainWindow::actionStart`

Definition at line 62 of file ui_mainwindow.h.

7.200.3.57 actionStep `QAction * Ui_MainWindow::actionStep`

Definition at line 63 of file ui_mainwindow.h.

7.200.3.58 actionStop `QAction * Ui_MainWindow::actionStop`

Definition at line 64 of file ui_mainwindow.h.

7.200.3.59 actionSupport `QAction * Ui_MainWindow::actionSupport`

Definition at line 73 of file ui_mainwindow.h.

7.200.3.60 actionTop `QAction * Ui_MainWindow::actionTop`

Definition at line 75 of file ui_mainwindow.h.

7.200.3.61 actionTrace `QAction * Ui_MainWindow::actionTrace`

Definition at line 81 of file ui_mainwindow.h.

7.200.3.62 actionUndo `QAction * Ui_MainWindow::actionUndo`

Definition at line 43 of file ui_mainwindow.h.

7.200.3.63 actionZoom_Facot `QAction * Ui_MainWindow::actionZoom_Facot`

Definition at line 91 of file ui_mainwindow.h.

7.200.3.64 actionZoom_Factor `QAction * Ui_MainWindow::actionZoom_Factor`

Definition at line 53 of file ui_mainwindow.h.

7.200.3.65 actionZoom_In `QAction * Ui_MainWindow::actionZoom_In`

Definition at line 51 of file ui_mainwindow.h.

7.200.3.66 actionZoom_In_2 `QAction * Ui_MainWindow::actionZoom_In_2`

Definition at line 89 of file ui_mainwindow.h.

7.200.3.67 actionZoom_Out `QAction * Ui_MainWindow::actionZoom_Out`

Definition at line 52 of file ui_mainwindow.h.

7.200.3.68 actionZoom_Out_2 `QAction * Ui_MainWindow::actionZoom_Out_2`

Definition at line 90 of file ui_mainwindow.h.

7.200.3.69 centralWidget `QWidget * Ui_MainWindow::centralWidget`

Definition at line 105 of file ui_mainwindow.h.

7.200.3.70 dockWidgetConsole `QDockWidget * Ui_MainWindow::dockWidgetConsole`

Definition at line 130 of file ui_mainwindow.h.

7.200.3.71 dockWidgetContents_Console `QWidget * Ui_MainWindow::dockWidgetContents_Console`

Definition at line 131 of file ui_mainwindow.h.

7.200.3.72 dockWidgetContents_Experiments `QWidget * Ui_MainWindow::dockWidgetContents_↩ Experiments`

Definition at line 139 of file ui_mainwindow.h.

7.200.3.73 dockWidgetContents_Properties `QWidget * Ui_MainWindow::dockWidgetContents_Properties`

Definition at line 127 of file ui_mainwindow.h.

7.200.3.74 dockWidgetContents_Reports `QWidget * Ui_MainWindow::dockWidgetContents_Reports`

Definition at line 135 of file ui_mainwindow.h.

7.200.3.75 dockWidgetExperiments `QDockWidget * Ui_MainWindow::dockWidgetExperiments`

Definition at line 138 of file ui_mainwindow.h.

7.200.3.76 dockWidgetPlugins `QDockWidget * Ui_MainWindow::dockWidgetPlugins`

Definition at line 122 of file ui_mainwindow.h.

7.200.3.77 dockWidgetPlugins_2 `QWidget * Ui_MainWindow::dockWidgetPlugins_2`

Definition at line 123 of file ui_mainwindow.h.

7.200.3.78 dockWidgetProperties `QDockWidget * Ui_MainWindow::dockWidgetProperties`

Definition at line 126 of file ui_mainwindow.h.

7.200.3.79 dockWidgetReports `QDockWidget * Ui_MainWindow::dockWidgetReports`

Definition at line 134 of file ui_mainwindow.h.

7.200.3.80 graphicsView `QGraphicsView * Ui_MainWindow::graphicsView`

Definition at line 110 of file ui_mainwindow.h.

7.200.3.81 gridLayout `QGridLayout * Ui_MainWindow::gridLayout`

Definition at line 106 of file ui_mainwindow.h.

7.200.3.82 horizontalLayout `QHBoxLayout * Ui_MainWindow::horizontalLayout`

Definition at line 109 of file ui_mainwindow.h.

7.200.3.83 horizontalLayout_2 `QHBoxLayout * Ui_MainWindow::horizontalLayout_2`

Definition at line 112 of file ui_mainwindow.h.

7.200.3.84 listWidget_Experiments `QListWidget * Ui_MainWindow::listWidget_Experiments`

Definition at line 140 of file ui_mainwindow.h.

7.200.3.85 listWidget_Plugins `QListWidget * Ui_MainWindow::listWidget_Plugins`

Definition at line 125 of file ui_mainwindow.h.

7.200.3.86 menuBar `QMenuBar * Ui_MainWindow::menuBar`

Definition at line 113 of file ui_mainwindow.h.

7.200.3.87 menuEdit `QMenu * Ui_MainWindow::menuEdit`

Definition at line 114 of file ui_mainwindow.h.

7.200.3.88 menuHelp `QMenu * Ui_MainWindow::menuHelp`

Definition at line 118 of file ui_mainwindow.h.

7.200.3.89 menuModel `QMenu * Ui_MainWindow::menuModel`

Definition at line 115 of file ui_mainwindow.h.

7.200.3.90 menuSimulator `QMenu * Ui_MainWindow::menuSimulator`

Definition at line 119 of file ui_mainwindow.h.

7.200.3.91 menuView_2 `QMenu * Ui_MainWindow::menuView_2`

Definition at line 116 of file ui_mainwindow.h.

7.200.3.92 menuWindow `QMenu * Ui_MainWindow::menuWindow`

Definition at line 117 of file ui_mainwindow.h.

7.200.3.93 statusBar `QStatusBar * Ui_MainWindow::statusBar`

Definition at line 120 of file ui_mainwindow.h.

7.200.3.94 tableView_Properties `QTableView * Ui_MainWindow::tableView_Properties`

Definition at line 129 of file ui_mainwindow.h.

7.200.3.95 tabModel1 `QWidget * Ui_MainWindow::tabModel1`

Definition at line 108 of file ui_mainwindow.h.

7.200.3.96 tabModel2 `QWidget * Ui_MainWindow::tabModel2`

Definition at line 111 of file ui_mainwindow.h.

7.200.3.97 tabWidgetModels `QTabWidget * Ui_MainWindow::tabWidgetModels`

Definition at line 107 of file ui_mainwindow.h.

7.200.3.98 textEdit_Console `QTextEdit * Ui_MainWindow::textEdit_Console`

Definition at line 133 of file ui_mainwindow.h.

7.200.3.99 textEdit_Report `QTextEdit * Ui_MainWindow::textEdit_Report`

Definition at line 137 of file ui_mainwindow.h.

7.200.3.100 toolBar `QToolBar * Ui_MainWindow::toolBar`

Definition at line 121 of file ui_mainwindow.h.

7.200.3.101 verticalLayout `QVBoxLayout * Ui_MainWindow::verticalLayout`

Definition at line 124 of file ui_mainwindow.h.

7.200.3.102 verticalLayout_3 `QVBoxLayout * Ui_MainWindow::verticalLayout_3`

Definition at line 128 of file ui_mainwindow.h.

7.200.3.103 verticalLayout_4 `QVBoxLayout * Ui_MainWindow::verticalLayout_4`

Definition at line 132 of file ui_mainwindow.h.

7.200.3.104 verticalLayout_5 `QVBoxLayout * Ui_MainWindow::verticalLayout_5`

Definition at line 136 of file ui_mainwindow.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/applications/gui/qt/mainWindow/ui_mainwindow.h

7.201 Unstore Class Reference

```
#include <Unstore.h>
```

Inheritance diagram for Unstore:

Collaboration diagram for Unstore:

Public Member Functions

- Unstore (Model ∗model, std::string name="")
- virtual ∼Unstore ()=default
- virtual std::string show ()

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.201.1 Detailed Description

Unstore module DESCRIPTION The Unstore module removes an entity from storage. When an entity arrives at the Unstore module, the storage specified is decreased and the entity immediately moves to the next module in the model. TYPICAL USES Removing the entity from an animation location when processing is complete Tracking the number of customers within a grocery store (unstore upon exit) PROMPTS Prompt Description Name Unique module identifier displayed on the module shape. Type Method of specifying the storage name as a Storage, Set, Attribute, or Expression. Default will remove an entity from the last storage that it entered. Storage Name Name of the storage to which the entity will be added. Applies only when the Type is Storage. Set Name Name of the storage set from which the storage is to be selected. Applies only when the Type is Set. Set Index Index into the defined storage set that contains the desired storage name. Applies only when the Type is Set. Attribute Name of the attribute whose value contains the storage. Applies only when the Type is Attribute. Expression Expression that is evaluated to the storage into which the entity is placed. Applies only when the Type is Expression.

Definition at line 45 of file Unstore.h.

7.201.2 Constructor & Destructor Documentation

7.201.2.1 Unstore()

```
Unstore::Unstore (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Unstore.cpp.

Here is the caller graph for this function:

7.201.2.2 ~Unstore()

```
virtual Unstore::~Unstore ( ) [virtual], [default]
```

7.201.3 Member Function Documentation

7.201.3.1 _check()

```
bool Unstore::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 56 of file Unstore.cpp.

7.201.3.2 _execute()

```
void Unstore::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 34 of file Unstore.cpp.

Here is the call graph for this function:

7.201.3.3 _initBetweenReplications()

```
void Unstore::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 47 of file Unstore.cpp.

7.201.3.4 _loadInstance()

```
bool Unstore::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelComponent.

Definition at line 39 of file Unstore.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.201.3.5 _saveInstance() `std::map< std::string, std::string > * Unstore::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 50 of file Unstore.cpp.

Here is the call graph for this function:

7.201.3.6 GetPluginInformation() `PluginInformation * Unstore::GetPluginInformation ( ) [static]`

Definition at line 63 of file Unstore.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.201.3.7 LoadInstance()

```
ModelComponent * Unstore::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 24 of file Unstore.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.201.3.8 show()

```
std::string Unstore::show ( ) [virtual]
```

Reimplemented from ModelComponent.

Definition at line 20 of file Unstore.cpp.

Here is the call graph for this function:

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Unstore.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Unstore.cpp

7.202 Util Class Reference

```
#include <Util.h>
```

Collaboration diagram for Util:

Public Types

- enum class TimeUnit : int {
  picosecond = 1 , nanosecond = 2 , microsecond = 3 , milisecond = 4 ,
  second = 5 , minute = 6 , hour = 7 , day = 8 ,
  week = 9 }
- enum class TraceLevel : int {
  L0_noTraces = 0 , L1_errorFatal = 1 , L2_results = 2 , L3_errorRecover = 3 ,
  L4_warning = 4 , L5_event = 5 , L6_arrival = 6 , L7_internal = 7 ,
  L8_detailed = 8 , L9_mostDetailed = 9 }
- typedef unsigned long identification
- typedef unsigned int rank

Static Public Member Functions

- static void SetIndent (const unsigned short indent)
- static void IncIndent ()
- static void DecIndent ()
- static void SepKeyVal (std::string str, std::string *key, std::string *value)
- static std::string Indent ()
- static std::string SetW (std::string text, unsigned short width)
- static std::string StrTimeUnitShort (Util::TimeUnit timeUnit)
- static std::string StrTimeUnitLong (Util::TimeUnit timeUnit)
- static Util::identification GenerateNewId ()
- static Util::identification GenerateNewIdOfType (std::string objtype)
- static Util::identification GetLastIdOfType (std::string objtype)
- static void ResetIdOfType (std::string objtype)
- static double TimeUnitConvert (Util::TimeUnit timeUnit1, Util::TimeUnit timeUnit2)
- template<class T >
  static std::string TypeOf ()
- template<class T >
  static Util::identification GenerateNewIdOfType ()

Static Public Attributes

- static unsigned int _S_indentation

7.202.1 Detailed Description

Definition at line 129 of file Util.h.

7.202.2 Member Typedef Documentation

7.202.2.1 identification `typedef unsigned long Util::identification`

Definition at line 131 of file Util.h.

7.202.2.2 rank `typedef unsigned int Util::rank`

Definition at line 132 of file Util.h.

7.202.3 Member Enumeration Documentation

7.202.3.1 TimeUnit `enum Util::TimeUnit : int [strong]`

Enumerator

picosecond	
nanosecond	
microsecond	
milisecond	
second	
minute	
hour	
day	
week	

Definition at line 134 of file Util.h.

7.202.3.2 TraceLevel `enum Util::TraceLevel : int [strong]`

Enumerator

L0_noTraces	
L1_errorFatal	
L2_results	
L3_errorRecover	
L4_warning	
L5_event	
L6_arrival	
L7_internal	
L8_detailed	
L9_mostDetailed	

Definition at line 147 of file Util.h.

7.202.4 Member Function Documentation

7.202.4.1 DecIndent() `void Util::DecIndent ( ) [static]`

Definition at line 33 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.2 GenerateNewId() `Util::identification Util::GenerateNewId ( ) [static]`

Definition at line 103 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.3 GenerateNewIdOfType() [1/2] template<class T >

static Util::identification Util::GenerateNewIdOfType ( ) [inline], [static]

Every component or element has a unique ID for its class, but not unique for other classes. IDs are generated sequentially for each class.

Definition at line 231 of file Util.h.

Here is the caller graph for this function:

7.202.4.4 GenerateNewIdOfType() [2/2] Util::identification Util::GenerateNewIdOfType (std::string *objtype*) [static]

Definition at line 108 of file Util.cpp.

7.202.4.5 GetLastIdOfType() Util::identification Util::GetLastIdOfType (std::string *objtype*) [static]

Definition at line 121 of file Util.cpp.

7.202.4.6 IncIndent() void Util::IncIndent ( ) [static]

Definition at line 25 of file Util.cpp.

7.202.4.7 Indent() `std::string Util::Indent ( ) [static]`

Definition at line 58 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.8 ResetIdOfType() `void Util::ResetIdOfType ( std::string objtype ) [static]`

Definition at line 135 of file Util.cpp.

7.202.4.9 SepKeyVal() `void Util::SepKeyVal ( std::string str, std::string * key, std::string * value ) [static]`

Definition at line 37 of file Util.cpp.

7.202.4.10 SetIndent() `void Util::SetIndent ( const unsigned short indent ) [static]`

Definition at line 29 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.11 SetW()

```
std::string Util::SetW (
    std::string text,
    unsigned short width ) [static]
```

Definition at line 66 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.12 StrTimeUnitLong()

```
std::string Util::StrTimeUnitLong (
    Util::TimeUnit timeUnit ) [static]
```

Definition at line 88 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.13 StrTimeUnitShort()

```
std::string Util::StrTimeUnitShort (
    Util::TimeUnit timeUnit ) [static]
```

Definition at line 72 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.14 TimeUnitConvert()

```
double Util::TimeUnitConvert (
    Util::TimeUnit timeUnit1,
    Util::TimeUnit timeUnit2 ) [static]
```

Definition at line 145 of file Util.cpp.

Here is the caller graph for this function:

7.202.4.15 TypeOf() `template<class T >`

`static std::string Util::TypeOf ( ) [inline], [static]`

Return the name of the class used as T.

Definition at line 214 of file Util.h.

7.202.5 Member Data Documentation

7.202.5.1 _S_indentation `unsigned int Util::_S_indentation [static]`

Definition at line 192 of file Util.h.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/util/Util.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/kernel/util/Util.cpp

7.203 Variable Class Reference

```
#include <Variable.h>
```

Inheritance diagram for Variable:

Collaboration diagram for Variable:

Public Member Functions

- Variable (Model ∗model, std::string name="")
- virtual ∼Variable ()=default
- virtual std::string show ()
- double value ()
- void setValue (double value)
- double value (std::string index)
- void setValue (std::string index, double value)
- double initialValue ()
- void setInitialValue (double value)
- double initialValue (std::string index)
- void setInitialValue (std::string index, double value)
- List< unsigned int > ∗ dimensionSizes () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelElement ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual bool _loadInstance (std::map< std::string, std::string > *fields)
- virtual std::map< std::string, std::string > * _saveInstance ()
- virtual bool _check (std::string *errorMessage)
- virtual void _initBetweenReplications ()

Additional Inherited Members

7.203.1 Detailed Description

Variable module DESCRIPTION This data module is used to define a variable's dimension and values. You can reference variables in other modules (for example, the Decide module), reassign new values to variables with the Assign module, and use variables in any expression. You can use an external data file to specify variable values, and you can specify the variable's initial values in the Variable module. If you use both methods, the values are read at different times, depending on the options you specify, including the File Read Time, the Clear Option, and the replication parameters you specify in the Run Setup dialog box. For more information, see the online Help. There are three methods for manually editing the Initial Values of a Variable module: Using the standard spreadsheet interface. In the module spreadsheet, right-click on the Initial Values cell and select the Edit via spreadsheet menu item. The values for two-dimensional arrays should be entered one column at a time. Array elements not explicitly assigned are assumed to have the last entered value. Using the module dialog box. In the module spreadsheet, right-click on any cell and select the Edit via dialog menu item. The values for two-dimensional arrays should be entered one column at a time. Array elements not explicitly assigned are assumed to have the last entered value. Using the two-dimensional (2-D) spreadsheet interface. In the module spreadsheet, click on the Initial Values cell. TYPICAL USES Number of documents processed per hour Serial number to assign to parts for unique identification Space available in a facility PROMPTS Prompt Description Name The unique name of the variable being defined. Rows Number of rows in a one- or two-dimensional variable. Columns Number of columns in a two-dimensional variable. Report Statistics Check box for determining whether or not statistics will be collected. This field is visible when the rows and columns are not specified (that is, for single variables). Data Type The data type of the values stored in the variable. Valid types are Real and String. The default type is Real. Clear Option Defines the time (if at all) when the value(s) of the variable is reset to the initial value(s) specified. Specifying Statistics resets this variable to its initial value(s) whenever statistics are cleared. Specifying System resets this variable to its initial value(s) whenever the system is cleared. Specifying None indicates that this variable is never reset to its initial value(s), except prior to the first replication. File Name Name of the file from which to read the variable's value or values. You can use any file access type supported by Arena except sequential text files and Lotus spreadsheet (.wks) files. If the file name you specify has not been created yet, Arena will create it, but you must edit the file to specify the file access type, path, and recordset (if required). Recordset Name of the recordset in the specified file from which to read values. This field is available only if you specify a File Name for a file that has been set up with a file access type, path, and recordset. Arena uses the Rows and Columns properties to determine the amount of data to read from the recordset. A recordset is required for all file types except .xml. The recordset size must be equal to or greater than the number of rows and columns specified for the variable. File Read Time Specifies when to read the values from the file into the variable. If you select PreCheck, the values for the variable are read while the model is still in Edit mode (prior to the model being checked and compiled). If you select BeginSimulation, values are read when the model is compiled, prior to the first replication. If you select BeginReplication, values are read prior to each replication. Initial Values Lists the initial value or values of the variable. You can assign new values to the variable at different stages of the model by using the Assign module. Initial Value Variable value at the start of the simulation.

Definition at line 90 of file Variable.h.

7.203.2 Constructor & Destructor Documentation

7.203.2.1 Variable()

```
Variable::Variable (
    Model * model,
    std::string name = "" )
```

Definition at line 17 of file Variable.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.203.2.2 ~Variable()

```
virtual Variable::~Variable ( ) [virtual], [default]
```

7.203.3 Member Function Documentation

7.203.3.1 _check()

```
bool Variable::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 134 of file Variable.cpp.

7.203.3.2 _initBetweenReplications()

```
void Variable::_initBetweenReplications ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 139 of file Variable.cpp.

7.203.3.3 _loadInstance()

```
bool Variable::_loadInstance (
    std::map< std::string, std::string > * fields ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 104 of file Variable.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.203.3.4 _saveInstance()

```
std::map< std::string, std::string > * Variable::_saveInstance ( ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 121 of file Variable.cpp.

Here is the call graph for this function:

7.203.3.5 dimensionSizes() `List< unsigned int > * Variable::dimensionSizes ( ) const`

Definition at line 90 of file Variable.cpp.

Here is the caller graph for this function:

7.203.3.6 GetPluginInformation() `PluginInformation * Variable::GetPluginInformation ( ) [static]`

Definition at line 31 of file Variable.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.203.3.7 initialValue() [1/2] `double Variable::initialValue ( )`

Definition at line 63 of file Variable.cpp.

7.203.3.8 initialValue() [2/2]

```
double Variable::initialValue (
    std::string index )
```

Definition at line 71 of file Variable.cpp.

7.203.3.9 LoadInstance()

```
ModelElement * Variable::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 94 of file Variable.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.203.3.10 setInitialValue() [1/2]

```
void Variable::setInitialValue (
    double value )
```

Definition at line 67 of file Variable.cpp.

Here is the call graph for this function:

7.203.3.11 setInitialValue() [2/2]

```
void Variable::setInitialValue (
    std::string index,
    double value )
```

Definition at line 80 of file Variable.cpp.

Here is the call graph for this function:

7.203.3.12 setValue() [1/2]

```
void Variable::setValue (
    double value )
```

Definition at line 49 of file Variable.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.203.3.13 setValue() [2/2]

```
void Variable::setValue (
    std::string index,
    double value )
```

Definition at line 53 of file Variable.cpp.

Here is the call graph for this function:

7.203.3.14 show()

```
std::string Variable::show ( ) [virtual]
```

Reimplemented from ModelElement.

Definition at line 21 of file Variable.cpp.

Here is the call graph for this function:

7.203.3.15 value() [1/2]

```
double Variable::value ( )
```

Definition at line 36 of file Variable.cpp.

Here is the caller graph for this function:

7.203.3.16 value() [2/2]

```
double Variable::value (
    std::string index )
```

Definition at line 40 of file Variable.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Variable.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Variable.cpp

7.204 Waiting Class Reference

```
#include <Queue.h>
```

Inheritance diagram for Waiting:

Collaboration diagram for Waiting:

Public Member Functions

- Waiting (Entity *entity, double timeStartedWaiting, ModelComponent *component, unsigned int component↩ OutputNumber=0)
- virtual ~Waiting ()=default
- virtual std::string show ()
- double getTimeStartedWaiting () const
- ModelComponent * getComponent () const
- Entity * getEntity () const
- unsigned int getComponentOutputNumber () const

7.204.1 Detailed Description

Definition at line 25 of file Queue.h.

7.204.2 Constructor & Destructor Documentation

7.204.2.1 Waiting()

```
Waiting::Waiting (
    Entity * entity,
    double timeStartedWaiting,
    ModelComponent * component,
    unsigned int componentOutputNumber = 0 )  [inline]
```

Definition at line 27 of file Queue.h.

7.204.2.2 ~Waiting() `virtual Waiting::~Waiting ( ) [virtual], [default]`

7.204.3 Member Function Documentation

7.204.3.1 getComponent() `ModelComponent* Waiting::getComponent ( ) const [inline]`

Definition at line 46 of file Queue.h.

7.204.3.2 getComponentOutputNumber() `unsigned int Waiting::getComponentOutputNumber ( ) const [inline]`

Definition at line 52 of file Queue.h.

7.204.3.3 getEntity() `Entity* Waiting::getEntity ( ) const [inline]`

Definition at line 49 of file Queue.h.

Here is the caller graph for this function:

7.204.3.4 getTimeStartedWaiting() `double Waiting::getTimeStartedWaiting ( ) const [inline]`

Definition at line 43 of file Queue.h.

Here is the caller graph for this function:

7.204.3.5 show() `virtual std::string Waiting::show ( ) [inline], [virtual]`

Reimplemented in WaitingResource.

Definition at line 36 of file Queue.h.

Here is the call graph for this function:

Here is the caller graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/elements/Queue.h

7.205 WaitingResource Class Reference

```
#include <Seize.h>
```

Inheritance diagram for WaitingResource:

Collaboration diagram for WaitingResource:

Public Member Functions

- WaitingResource (Entity *entity, double timeStartedWaiting, unsigned int quantity, ModelComponent *component)
- WaitingResource (const WaitingResource &orig)
- virtual ~WaitingResource ()=default
- virtual std::string show ()
- unsigned int getQuantity () const

7.205.1 Detailed Description

Definition at line 26 of file Seize.h.

7.205.2 Constructor & Destructor Documentation

7.205.2.1 WaitingResource() [1/2]

```
WaitingResource::WaitingResource (
    Entity * entity,
    double timeStartedWaiting,
    unsigned int quantity,
    ModelComponent * component ) [inline]
```

Definition at line 28 of file Seize.h.

7.205.2.2 WaitingResource() [2/2]

```
WaitingResource::WaitingResource (
    const WaitingResource & orig ) [inline]
```

Definition at line 31 of file Seize.h.

7.205.2.3 ~WaitingResource()

```
virtual WaitingResource::~WaitingResource ( ) [virtual], [default]
```

7.205.3 Member Function Documentation

7.205.3.1 getQuantity()

```
unsigned int WaitingResource::getQuantity ( ) const [inline]
```

Definition at line 41 of file Seize.h.

7.205.3.2 show()

```
virtual std::string WaitingResource::show ( ) [inline], [virtual]
```

Reimplemented from Waiting.

Definition at line 36 of file Seize.h.

Here is the call graph for this function:

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Seize.h

7.206 Write Class Reference

```
#include <Write.h>
```

Inheritance diagram for Write:

Collaboration diagram for Write:

Public Types

- enum class WriteToType : int { SCREEN = 1 , FILE = 2 }

Public Member Functions

- Write (Model ∗model, std::string name="")
- virtual ∼Write ()=default
- virtual std::string show ()
- List< WriteText ∗ > ∗ writeElements () const
- void setFilename (std::string _filename)
- std::string filename () const
- void setWriteToType (WriteToType _writeToType)
- Write::WriteToType writeToType () const

Static Public Member Functions

- static PluginInformation ∗ GetPluginInformation ()
- static ModelComponent ∗ LoadInstance (Model ∗model, std::map< std::string, std::string > ∗fields)

Protected Member Functions

- virtual void _execute (Entity ∗entity)
- virtual void _initBetweenReplications ()
- virtual bool _loadInstance (std::map< std::string, std::string > ∗fields)
- virtual std::map< std::string, std::string > ∗ _saveInstance ()
- virtual bool _check (std::string ∗errorMessage)

Additional Inherited Members

7.206.1 Detailed Description

This component ...

Definition at line 80 of file Write.h.

7.206.2 Member Enumeration Documentation

7.206.2.1 WriteToType `enum Write::WriteToType : int [strong]`

Enumerator

SCREEN	
FILE	

Definition at line 83 of file Write.h.

7.206.3 Constructor & Destructor Documentation

7.206.3.1 Write()

```
Write::Write (
    Model * model,
    std::string name = "" )
```

Definition at line 19 of file Write.cpp.

Here is the caller graph for this function:

7.206.3.2 ~Write() `virtual Write::~Write ( ) [virtual], [default]`

7.206.4 Member Function Documentation

7.206.4.1 _check()

```
bool Write::_check (
    std::string * errorMessage ) [protected], [virtual]
```

Reimplemented from ModelElement.

Definition at line 138 of file Write.cpp.

Here is the call graph for this function:

7.206.4.2 _execute()

```
void Write::_execute (
    Entity * entity ) [protected], [virtual]
```

Implements ModelComponent.

Definition at line 56 of file Write.cpp.

Here is the call graph for this function:

7.206.4.3 _initBetweenReplications() `void Write::_initBetweenReplications ( ) [protected], [virtual]`

Reimplemented from ModelElement.

Definition at line 84 of file Write.cpp.

Here is the call graph for this function:

7.206.4.4 _loadInstance() `bool Write::_loadInstance ( std::map< std::string, std::string > * fields ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 95 of file Write.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.206.4.5 _saveInstance() `std::map< std::string, std::string > * Write::_saveInstance ( ) [protected], [virtual]`

Reimplemented from ModelComponent.

Definition at line 122 of file Write.cpp.

Here is the call graph for this function:

7.206.4.6 filename() `std::string Write::filename ( ) const`

Definition at line 44 of file Write.cpp.

7.206.4.7 GetPluginInformation() `PluginInformation * Write::GetPluginInformation ( ) [static]`

Definition at line 156 of file Write.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.206.4.8 LoadInstance()

```
ModelComponent * Write::LoadInstance (
    Model * model,
    std::map< std::string, std::string > * fields ) [static]
```

Definition at line 26 of file Write.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

7.206.4.9 setFilename()

```
void Write::setFilename (
    std::string _filename )
```

Definition at line 40 of file Write.cpp.

7.206.4.10 setWriteToType()

```
void Write::setWriteToType (
    WriteToType _writeToType )
```

Definition at line 48 of file Write.cpp.

Here is the caller graph for this function:

7.206.4.11 show() `std::string Write::show ( ) [virtual]`

Reimplemented from ModelComponent.

Definition at line 22 of file Write.cpp.

Here is the call graph for this function:

7.206.4.12 writeElements() `List< WriteText * > * Write::writeElements ( ) const`

Definition at line 36 of file Write.cpp.

Here is the caller graph for this function:

7.206.4.13 writeToType() `Write::WriteToType Write::writeToType ( ) const`

Definition at line 52 of file Write.cpp.

The documentation for this class was generated from the following files:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h
- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.cpp

7.207 WriteExpression Class Reference

```
#include <Write.h>
```

Inheritance diagram for WriteExpression:

Collaboration diagram for WriteExpression:

Classes

- struct DEFAULT_VALUES

Public Member Functions

- WriteExpression (std::string text="")

Public Attributes

- const struct WriteExpression::DEFAULT_VALUES DEFAULT

7.207.1 Detailed Description

Definition at line 50 of file Write.h.

7.207.2 Constructor & Destructor Documentation

7.207.2.1 WriteExpression()

```
WriteExpression::WriteExpression (
    std::string text = "" ) [inline]
```

Definition at line 52 of file Write.h.

7.207.3 Member Data Documentation

7.207.3.1 DEFAULT `const struct WriteExpression::DEFAULT_VALUES WriteExpression::DEFAULT`

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.208 WriteInExpression Class Reference

```
#include <Write.h>
```

Inheritance diagram for WriteInExpression:

Collaboration diagram for WriteInExpression:

Classes

- struct DEFAULT_VALUES

Public Member Functions

- WriteInExpression (std::string text="")

Public Attributes

- const struct WriteInExpression::DEFAULT_VALUES DEFAULT

7.208.1 Detailed Description

Definition at line 63 of file Write.h.

7.208.2 Constructor & Destructor Documentation

7.208.2.1 WriteInExpression() `WriteInExpression::WriteInExpression ( std::string text = "" ) [inline]`

Definition at line 65 of file Write.h.

7.208.3 Member Data Documentation

7.208.3.1 DEFAULT `const struct WritelnExpression::DEFAULT_VALUES WritelnExpression::DEFAULT`

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.209 WriteInText Class Reference

```
#include <Write.h>
```

Inheritance diagram for WriteInText:

Collaboration diagram for WriteInText:

Classes

- struct DEFAULT_VALUES

Public Member Functions

- WriteInText (std::string text="")

Public Attributes

- const struct WriteInText::DEFAULT_VALUES DEFAULT

7.209.1 Detailed Description

Definition at line 37 of file Write.h.

7.209.2 Constructor & Destructor Documentation

7.209.2.1 WriteInText()

```
WriteInText::WriteInText (
    std::string text = "" ) [inline]
```

Definition at line 39 of file Write.h.

7.209.3 Member Data Documentation

7.209.3.1 DEFAULT `const struct WritelnText::DEFAULT_VALUES WritelnText::DEFAULT`

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

7.210 WriteText Class Reference

```
#include <Write.h>
```

Inheritance diagram for WriteText:

Collaboration diagram for WriteText:

Classes

- struct DEFAULT_VALUES

Public Member Functions

- WriteText (std::string text="")

Public Attributes

- const struct WriteText::DEFAULT_VALUES DEFAULT
- std::string text = DEFAULT.text
- bool isExpression = DEFAULT.isExpression
- bool newline = DEFAULT.newline

7.210.1 Detailed Description

Definition at line 19 of file Write.h.

7.210.2 Constructor & Destructor Documentation

7.210.2.1 WriteText() `WriteText::WriteText ( std::string text = "" ) [inline]`

Definition at line 21 of file Write.h.

7.210.3 Member Data Documentation

7.210.3.1 DEFAULT `const struct WriteText::DEFAULT_VALUES WriteText::DEFAULT`

7.210.3.2 isExpression `bool WriteText::isExpression = DEFAULT.isExpression`

Definition at line 33 of file Write.h.

7.210.3.3 newline `bool WriteText::newline = DEFAULT.newline`

Definition at line 34 of file Write.h.

7.210.3.4 text `std::string WriteText::text = DEFAULT.text`

Definition at line 32 of file Write.h.

The documentation for this class was generated from the following file:

- /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_↩ 2022_GenESyS/plugins/components/Write.h

8 File Documentation

8.1 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/.dep.inc File Reference

8.2 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/.dep.inc

```
# This code depends on make tool being used
DEPFILES=$(wildcard $(addsuffix .d, ${OBJECTFILES} ${TESTOBJECTFILES}))
ifneq (${DEPFILES},)
include ${DEPFILES}
endif
```

8.3 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/BaseConsoleGenesys↩ Application.cpp File Reference

```
#include "BaseConsoleGenesysApplication.h"
#include <iostream>
#include "../kernel/simulator/Simulator.h"
```

Include dependency graph for BaseConsoleGenesysApplication.cpp:

8.4 BaseConsoleGenesysApplication.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   BaseConsoleGenesysApplication.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 16:25
 */

#include "BaseConsoleGenesysApplication.h"
#include <iostream>

#include "../kernel/simulator/Simulator.h"

BaseConsoleGenesysApplication::BaseConsoleGenesysApplication() {
}


// default Trace Handlers

void BaseConsoleGenesysApplication::traceHandler(TraceEvent e) {
    std::cout « e.getText() « std::endl;
}

void BaseConsoleGenesysApplication::traceErrorHandler(TraceErrorEvent e) {
    std::cout « e.getText() « std::endl;
}

void BaseConsoleGenesysApplication::traceReportHandler(TraceEvent e) {
    std::cout « "" « e.getText() « "" « std::endl;
}

void BaseConsoleGenesysApplication::traceSimulationHandler(TraceSimulationEvent e) {
    std::cout « e.getText() « std::endl;
}


// default Event Handlers

```

```
00044 void BaseConsoleGenesysApplication::onBreakpointHandler(SimulationEvent* re) {
00045     std::cout « "(Event Handler) " « "Breakpoint occured at time " «
       std::to_string(re->getSimulatedTime()) « std::endl;
00046 }
00047
00048 void BaseConsoleGenesysApplication::onEntityCreateHandler(SimulationEvent* re) {
00049     std::cout « "(Event Handler) " « "Entity " « re->getEntityCreated()->getName() « " was created." «
       std::endl;
00050 }
00051
00052 void BaseConsoleGenesysApplication::onEntityMoveHandler(SimulationEvent* re) {
00053     std::cout « "(Event Handler) " « "Entity " « re->getCurrentEntity()->getName() « " was moved to
       component" « re->getDestinationComponent()->getName() « std::endl;
00054 }
00055
00056 void BaseConsoleGenesysApplication::onEntityRemoveHandler(SimulationEvent* re) {
00057     std::cout « "(Event Handler) " « "Entity " « re->getCurrentEvent()->getEntity()->getName() « " was
       removed." « std::endl;
00058 }
00059
00060 void BaseConsoleGenesysApplication::onProcessEventHandler(SimulationEvent* re) {
00061     std::cout « "(Event Handler) " « "Processing event " « re->getCurrentEvent()->show() « std::endl;
00062 }
00063
00064 void BaseConsoleGenesysApplication::onReplicationEndHandler(SimulationEvent* re) {
00065     std::cout « "(Event Handler) " « "Replication " « re->getCurrentReplicationNumber() « " ending." «
       std::endl;
00066 }
00067
00068 void BaseConsoleGenesysApplication::onReplicationStartHandler(SimulationEvent* re) {
00069     std::cout « "(Event Handler) " « "Replication " « re->getCurrentReplicationNumber() « " starting."
       « std::endl;
00070 }
00071
00072 void BaseConsoleGenesysApplication::onReplicationStepHandler(SimulationEvent* re) {
00073     std::cout « "(Event Handler) " « "Replication step at time " «
       std::to_string(re->getSimulatedTime()) « std::endl;
00074 }
00075
00076 void BaseConsoleGenesysApplication::onSimulationStartHandler(SimulationEvent* re) {
00077     std::cout « "(Event Handler) " « "Simulation is starting" « std::endl;
00078 }
00079
00080 void BaseConsoleGenesysApplication::onSimulationEndHandler(SimulationEvent* re) {
00081     std::cout « "Event (Handler) " « "Replication " « re->getCurrentReplicationNumber() « " ending." «
       std::endl;
00082 }
00083
00084 void BaseConsoleGenesysApplication::onSimulationPausedHandler(SimulationEvent* re) {
00085     std::cout « "(Event Handler) " « "Simulation was paused at time " «
       std::to_string(re->getSimulatedTime()) « std::endl;
00086 }
00087
00088 void BaseConsoleGenesysApplication::onSimulationResumeHandler(SimulationEvent* re) {
00089     std::cout « "(Event Handler) " « "Simulation was resumed." « std::endl;
00090 }
00091
00092 void BaseConsoleGenesysApplication::setDefaultEventHandlers(OnEventManager* oem) {
00093     oem->addOnBreakpointHandler(this, &BaseConsoleGenesysApplication::onBreakpointHandler);
00094     oem->addOnEntityCreateHandler(this, &BaseConsoleGenesysApplication::onEntityCreateHandler);
00095     oem->addOnEntityMoveHandler(this, &BaseConsoleGenesysApplication::onEntityMoveHandler);
00096     oem->addOnEntityRemoveHandler(this, &BaseConsoleGenesysApplication::onEntityRemoveHandler);
00097     oem->addOnProcessEventHandler(this, &BaseConsoleGenesysApplication::onProcessEventHandler);
00098     oem->addOnReplicationEndHandler(this, &BaseConsoleGenesysApplication::onReplicationEndHandler);
00099     oem->addOnReplicationStartHandler(this,
       &BaseConsoleGenesysApplication::onReplicationStartHandler);
00100     oem->addOnReplicationStepHandler(this, &BaseConsoleGenesysApplication::onReplicationStepHandler);
00101     oem->addOnSimulationEndHandler(this, &BaseConsoleGenesysApplication::onSimulationEndHandler);
00102     oem->addOnSimulationPausedHandler(this,
       &BaseConsoleGenesysApplication::onSimulationPausedHandler);
00103     oem->addOnSimulationResumeHandler(this,
       &BaseConsoleGenesysApplication::onSimulationResumeHandler);
00104     oem->addOnSimulationStartHandler(this, &BaseConsoleGenesysApplication::onSimulationStartHandler);
00105 }
00106
00107 void BaseConsoleGenesysApplication::setDefaultTraceHandlers(TraceManager* tm) {
00108     tm->addTraceHandler<BaseConsoleGenesysApplication>(this,
       &BaseConsoleGenesysApplication::traceHandler);
00109     tm->addTraceErrorHandler<BaseConsoleGenesysApplication>(this,
       &BaseConsoleGenesysApplication::traceErrorHandler);
00110     tm->addTraceReportHandler<BaseConsoleGenesysApplication>(this,
       &BaseConsoleGenesysApplication::traceReportHandler);
00111     tm->addTraceSimulationHandler<BaseConsoleGenesysApplication>(this,
       &BaseConsoleGenesysApplication::traceSimulationHandler);
00112 }
00113
00114 void BaseConsoleGenesysApplication::insertFakePluginsByHand(Simulator* simulator) {
```

```
    // model components
    // arena basic process
    simulator->getPlugins()->insert("create.so");
    simulator->getPlugins()->insert("dispose.so");
    simulator->getPlugins()->insert("decide.so");
    simulator->getPlugins()->insert("batch.so");
    simulator->getPlugins()->insert("separate.so");
    simulator->getPlugins()->insert("assign.so");
    simulator->getPlugins()->insert("record.so");
    simulator->getPlugins()->insert("process.so");
    simulator->getPlugins()->insert("submodel.so");
    simulator->getPlugins()->insert("entitygroup.so");
    simulator->getPlugins()->insert("queue.so");
    simulator->getPlugins()->insert("set.so");
    simulator->getPlugins()->insert("resource.so");
    simulator->getPlugins()->insert("variable.so");
    simulator->getPlugins()->insert("schedule.so");
    simulator->getPlugins()->insert("entitygroup.so");
    // arena advanced process
    simulator->getPlugins()->insert("delay.so");
    simulator->getPlugins()->insert("dropoff.so");
    simulator->getPlugins()->insert("hold.so");
    simulator->getPlugins()->insert("match.so");
    simulator->getPlugins()->insert("pickup.so");
    simulator->getPlugins()->insert("read.so");
    simulator->getPlugins()->insert("write.so");
    simulator->getPlugins()->insert("release.so");
    simulator->getPlugins()->insert("remove.so");
    simulator->getPlugins()->insert("seize.so");
    simulator->getPlugins()->insert("search.so");
    simulator->getPlugins()->insert("signal.so");
    simulator->getPlugins()->insert("store.so");
    simulator->getPlugins()->insert("unstore.so");
    simulator->getPlugins()->insert("expression.so");
    simulator->getPlugins()->insert("failure.so");
    simulator->getPlugins()->insert("file.so");
    simulator->getPlugins()->insert("storage.so");
    // arena transfer station
    simulator->getPlugins()->insert("enter.so");
    simulator->getPlugins()->insert("leave.so");
    simulator->getPlugins()->insert("pickstation.so");
    simulator->getPlugins()->insert("route.so");
    simulator->getPlugins()->insert("sequence.so");
    simulator->getPlugins()->insert("station.so");
    // arena transfer conveyour
    simulator->getPlugins()->insert("access.so");
    simulator->getPlugins()->insert("exit.so");
    simulator->getPlugins()->insert("start.so");
    simulator->getPlugins()->insert("stop.so");
    simulator->getPlugins()->insert("conveyour.so");
    simulator->getPlugins()->insert("segment.so");
    // arena transfer transport
    simulator->getPlugins()->insert("alocate.so");
    simulator->getPlugins()->insert("free.so");
    simulator->getPlugins()->insert("halt.so");
    simulator->getPlugins()->insert("move.so");
    simulator->getPlugins()->insert("request.so");
    simulator->getPlugins()->insert("transporter.so");
    simulator->getPlugins()->insert("distance.so");
    simulator->getPlugins()->insert("network.so");
    simulator->getPlugins()->insert("networklink.so");
    // others
    simulator->getPlugins()->insert("dummy.so");
    simulator->getPlugins()->insert("lsode.so");
    simulator->getPlugins()->insert("biochemical.so");
    simulator->getPlugins()->insert("markovchain.so");
    simulator->getPlugins()->insert("cellularautomata.so");
    simulator->getPlugins()->insert("cppforgenesys.so");
}
```

8.5 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/BaseConsoleGenesys↩ Application.h File Reference

```
#include "GenesysApplication_if.h"
#include "../kernel/simulator/TraceManager.h"
#include "../kernel/simulator/OnEventManager.h"
```

Include dependency graph for BaseConsoleGenesysApplication.h:

Classes

- class BaseConsoleGenesysApplication

8.6 BaseConsoleGenesysApplication.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   BaseConsoleGenesysApplication.h
00009  * Author: rlcancian
00010  *
00011  * Created on 3 de Setembro de 2019, 16:25
00012  */
00013
00014 #ifndef BASECONSOLEGENESYSAPPLICATION_H
00015 #define BASECONSOLEGENESYSAPPLICATION_H
00016
00017 #include "GenesysApplication_if.h"
00018 #include "../kernel/simulator/TraceManager.h"
00019 #include "../kernel/simulator/OnEventManager.h"
00020
00021 class BaseConsoleGenesysApplication : public GenesysApplication_if {
00022 public:
00023     BaseConsoleGenesysApplication();
00024     virtual ~BaseConsoleGenesysApplication() = default;
00025 public:
00026     virtual int main(int argc, char** argv) = 0;
00027 public:
00028     void setDefaultTraceHandlers(TraceManager* tm);
00029     void setDefaultEventHandlers(OnEventManager* oem);
00030     void insertFakePluginsByHand(Simulator* simulator);
00031 protected:
00032     // default Trace Handlers
00033     virtual void traceHandler(TraceEvent e);
00034     virtual void traceErrorHandler(TraceErrorEvent e);
00035     virtual void traceReportHandler(TraceEvent e);
00036     virtual void traceSimulationHandler(TraceSimulationEvent e);
00037     // default Event Handlers
00038     virtual void onBreakpointHandler(SimulationEvent* re);
00039     virtual void onEntityCreateHandler(SimulationEvent* re);
00040     virtual void onEntityMoveHandler(SimulationEvent* re);
```

```
    virtual void onSimulationStartHandler(SimulationEvent* re);
    virtual void onReplicationStepHandler(SimulationEvent* re);
    virtual void onReplicationStartHandler(SimulationEvent* re);
    virtual void onProcessEventHandler(SimulationEvent* re);
    virtual void onReplicationEndHandler(SimulationEvent* re);
    virtual void onSimulationEndHandler(SimulationEvent* re);
    virtual void onSimulationPausedHandler(SimulationEvent* re);
    virtual void onSimulationResumeHandler(SimulationEvent* re);
    virtual void onEntityRemoveHandler(SimulationEvent* re);
private:

};

#endif /* BASECONSOLEGENESYSAPPLICATION_H */

```

8.7 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/GenesysApplication_if.h File Reference

This graph shows which files directly or indirectly include this file:

Classes

- class GenesysApplication_if

8.8 GenesysApplication_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   GenesysApplication_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 2 de Outubro de 2018, 19:14
 */

#ifndef GENESYSAPPLICATION_IF_H
#define GENESYSAPPLICATION_IF_H

class GenesysApplication_if {
public:
    virtual int main(int argc, char** argv) = 0;
};

#endif /* GENESYSAPPLICATION_IF_H */

```

8.9 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/GenesysQtGUI.cpp File Reference

```
#include "GenesysQtGUI.h"
#include <QApplication>
```

```
#include "mainWindow/mainwindow.h"
```

Include dependency graph for GenesysQtGUI.cpp:

8.10 GenesysQtGUI.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   GenesysQtGUI.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 20 de maio de 2021, 14:37
00012  */
00013
00014 #include "GenesysQtGUI.h"
00015 #include <QApplication>
00016 #include "mainWindow/mainwindow.h"
00017
00018 GenesysQtGUI::GenesysQtGUI() {
00019 }
00020
00021 int GenesysQtGUI::main(int argc, char** argv) {
00022     QApplication app(argc, argv);
00023     MainWindow w;
00024     w.show();
00025     return app.exec();
00026 }
```

8.11 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/GenesysQtGUI.h File Reference

```
#include "../../GenesysApplication_if.h"
```

Include dependency graph for GenesysQtGUI.h:

This graph shows which files directly or indirectly include this file:

Classes

- class GenesysQtGUI

8.12 GenesysQtGUI.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   GenesysQtGUI.h
 * Author: rlcancian
 *
 * Created on 20 de maio de 2021, 14:37
 */

// no ubuntu, instalar libqt5webkit5-dev
// sudo apt-get install libqt5webkit5-dev

#ifndef GENESYSQTGUI_H
#define GENESYSQTGUI_H

#include "../../GenesysApplication_if.h"

class GenesysQtGUI : public GenesysApplication_if {
public:
```

```
    GenesysQtGUI();
public:
    virtual int main(int argc, char** argv);
private:

};

#endif /* GENESYSQTGUI_H */

```

8.13 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/main↩ Window/mainwindow.cpp File Reference

```
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include "../dialogs/dialogabout.h"
#include "../dialogs/dialogmodelinformation.h"
#include "Traits.h"
#include <Qt>
#include <QFileDialog>
#include <QMessageBox>
#include <QStandardItem>
#include <QListWidget>
#include <QListWidgetItem>
#include <QString>
#include <string>
#include <limits.h>
#include <unistd.h>
#include <errno.h>
#include <vector>
#include <iostream>
```

Include dependency graph for mainwindow.cpp:

8.14 mainwindow.cpp

```
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include "../dialogs/dialogabout.h"
#include "../dialogs/dialogmodelinformation.h"
#include "Traits.h"
#include <Qt>
#include <QFileDialog>
#include <QMessageBox>
#include <QStandardItem>
#include <QListWidget>
#include <QListWidgetItem>
#include <QString>

// get path
#include <string>
#include <limits.h>
#include <unistd.h>
#include <errno.h>
#include <vector>
#include <string>
#include <iostream>

MainWindow::MainWindow(QWidget *parent) :
QMainWindow(parent),
```

```
ui(new Ui::MainWindow) {
    ui->setupUi(this);
    connect(ui->actionAbout, SIGNAL(triggered(bool)), this, SLOT(on_actionAbout_triggered()));

    // action menu on plugin list
    auto actView = new QAction("View", this);
    // auto actDelete = new QAction("Delete", this);
    // you can set up slot connections here or in designer
    connect(actView, SIGNAL(triggered()), this, SLOT(viewItem()));
    //connect(actDelete, SIGNAL(triggered()), this, SLOT(eraseItem()));
    ui->listWidget_Plugins->setContextMenuPolicy(Qt::ActionsContextMenu);
    ui->listWidget_Plugins->addActions({actView}); //, actDelete});
    //--------------------------
    // models
    //ui->tabWidgetModels->clear(); // no models openned

    //*****************************************************************
    this->simulator = new Simulator();
    TraceManager* tm = simulator->getTracer();
    tm->setTraceLevel(Traits<GenesysApplication_if>::traceLevel);
    tm->addTraceHandler<MainWindow>(this, &MainWindow::_traceHandler);
    tm->addTraceReportHandler<MainWindow>(this, &MainWindow::_traceReportHandler);
    tm->addTraceSimulationHandler<MainWindow>(this, &MainWindow::_traceSimulationHandler);
    tm->addTraceErrorHandler<MainWindow>(this, &MainWindow::_traceErrorHandler);
    _simulationInsertPlugins();
    // refresh UI
    _refreshActions();
    _refreshTabWidgetModel();
    QString title = QString::fromStdString("Generic and Expansible System Simulator (" +
std::to_string(this->simulator->getVersionNumber()) + ")");
    this->setWindowTitle(title);
}

MainWindow::~MainWindow() {
    delete ui;
}

//*******************************************************************
// SIMULATOR START
//*******************************************************************

void MainWindow::_insertPluginUI(Plugin* plugin) {
    if (plugin != nullptr) {
        if (plugin->isIsValidPlugin()) {
            QListWidgetItem* item = new
QListWidgetItem(QString::fromStdString(plugin->getPluginInfo()->getPluginTypename()));
            std::string plugtextAdds = "";
            if (plugin->getPluginInfo()->isComponent()) {
                plugtextAdds += ",Component";
                //item->setBackgroundColor(Qt::white);
                //item->set
                item->setIcon(QIcon(":/resources/icons/pack3/png/24x24/Component.png"));
            } else {
                plugtextAdds += ",Element";
                //item->setBackgroundColor(Qt::lightGray);
                item->setIcon(QIcon(":/resources/icons/pack3/png/24x24/Calendar.png"));
                //item->setFont(QFont::Style::StyleItalic);
            }
            if (plugin->getPluginInfo()->isSink()) {
                plugtextAdds += ",Sink";
                item->setIcon(QIcon(":/resources/icons/pack3/png/24x24/LoadInv.png"));
            }
            if (plugin->getPluginInfo()->isSource()) {
                plugtextAdds += ",Source";
                item->setIcon(QIcon(":/resources/icons/pack3/png/24x24/Load.png"));
            }
            if (plugin->getPluginInfo()->isReceiveTransfer()) {
                plugtextAdds += ",ReceiveTransfer";
                item->setIcon(QIcon(":/resources/icons/pack3/png/24x24/Load.png"));
            }
            if (plugin->getPluginInfo()->isSendTransfer()) {
                plugtextAdds += ",SendTransfer";
                item->setIcon(QIcon(":/resources/icons/pack3/png/24x24/LoadInv.png"));
            }
            plugtextAdds += "\n" + plugin->getPluginInfo()->getObservation(); //getDescriptionHelp();
            plugtextAdds = plugtextAdds.erase(0, 1);
            item->setToolTip(QString::fromStdString(plugtextAdds));
            item->setStatusTip(QString::fromStdString(plugtextAdds));
            ui->listWidget_Plugins->addItem(item);
        }
    }
}

void MainWindow::__simulationInsert_FAKE_Plugins() {
    //Util::TraceLevel tl = simulator->getTracer()->traceLevel();
    //simulator->getTracer()->setTraceLevel(Util::TraceLevel::debugOnly);
    PluginManager* plm = simulator->getPlugins();
```

```

    _insertPluginUI(plm->insert("create.so"));
    _insertPluginUI(plm->insert("dispose.so"));
    _insertPluginUI(plm->insert("decide.so"));
    _insertPluginUI(plm->insert("batch.so"));
    _insertPluginUI(plm->insert("separate.so"));
    _insertPluginUI(plm->insert("assign.so"));
    _insertPluginUI(plm->insert("record.so"));
    _insertPluginUI(plm->insert("submodel.so"));
    //_insertPluginUI(plm->insert("entitytype.so"));
    _insertPluginUI(plm->insert("entitygroup.so"));
    //_insertPluginUI(plm->insert("attribute.so"));
    //_insertPluginUI(plm->insert("counter.so"));
    _insertPluginUI(plm->insert("queue.so"));
    _insertPluginUI(plm->insert("set.so"));
    _insertPluginUI(plm->insert("resource.so"));
    _insertPluginUI(plm->insert("variable.so"));
    _insertPluginUI(plm->insert("schedule.so"));
    _insertPluginUI(plm->insert("entitygroup.so"));
    // arena advanced process
    _insertPluginUI(plm->insert("delay.so"));
    _insertPluginUI(plm->insert("dropoff.so"));
    _insertPluginUI(plm->insert("hold.so"));
    _insertPluginUI(plm->insert("match.so"));
    _insertPluginUI(plm->insert("pickup.so"));
    _insertPluginUI(plm->insert("read.so"));
    _insertPluginUI(plm->insert("write.so"));
    _insertPluginUI(plm->insert("release.so"));
    _insertPluginUI(plm->insert("remove.so"));
    _insertPluginUI(plm->insert("seize.so"));
    _insertPluginUI(plm->insert("search.so"));
    _insertPluginUI(plm->insert("signal.so"));
    _insertPluginUI(plm->insert("store.so"));
    _insertPluginUI(plm->insert("unstore.so"));
    _insertPluginUI(plm->insert("expression.so"));
    _insertPluginUI(plm->insert("failure.so"));
    _insertPluginUI(plm->insert("file.so"));
    _insertPluginUI(plm->insert("statisticscollector.so"));
    _insertPluginUI(plm->insert("storage.so"));
    // arena transfer station
    _insertPluginUI(plm->insert("enter.so"));
    _insertPluginUI(plm->insert("leave.so"));
    _insertPluginUI(plm->insert("pickstation.so"));
    _insertPluginUI(plm->insert("route.so"));
    _insertPluginUI(plm->insert("sequence.so"));
    _insertPluginUI(plm->insert("station.so"));
    // arena transfer conveyour
    _insertPluginUI(plm->insert("access.so"));
    _insertPluginUI(plm->insert("exit.so"));
    _insertPluginUI(plm->insert("start.so"));
    _insertPluginUI(plm->insert("stop.so"));
    _insertPluginUI(plm->insert("conveyour.so"));
    _insertPluginUI(plm->insert("segment.so"));
    // arena transfer transport
    _insertPluginUI(plm->insert("alocate.so"));
    _insertPluginUI(plm->insert("free.so"));
    _insertPluginUI(plm->insert("halt.so"));
    _insertPluginUI(plm->insert("move.so"));
    _insertPluginUI(plm->insert("request.so"));
    _insertPluginUI(plm->insert("transporter.so"));
    _insertPluginUI(plm->insert("distance.so"));
    _insertPluginUI(plm->insert("network.so"));
    _insertPluginUI(plm->insert("networklink.so"));
    // others
    _insertPluginUI(plm->insert("dummy.so"));
    _insertPluginUI(plm->insert("lsode.so"));
    _insertPluginUI(plm->insert("biochemical.so"));
    _insertPluginUI(plm->insert("markovchain.so"));
    _insertPluginUI(plm->insert("cellularautomata.so"));
    _insertPluginUI(plm->insert("cppforgenesys.so"));
    //simulator->getTracer()->setTraceLevel(tl);
}

void MainWindow::_simulationInsertPlugins() {
    // TODO: Not implemented yet
    __simulationInsert_FAKE_Plugins();
}

void MainWindow::_traceHandler(TraceEvent e) {
    // TODO: -- use e.getTracelevel() to set text style (color, bold, etc)
    if (e.getTracelevel() == Util::TraceLevel::L1_errorFatal)
        ui->textEdit_Console->setTextColor(QColor::fromRgb(255, 0, 0));
    else if (e.getTracelevel() == Util::TraceLevel::L3_errorRecover)
        ui->textEdit_Console->setTextColor(QColor::fromRgb(128, 0, 0));
    else {
        unsigned short grayVal = 5 * (static_cast<unsigned int> (e.getTracelevel()));
        ui->textEdit_Console->setTextColor(QColor::fromRgb(grayVal, grayVal, grayVal));
```

```
    }
    ui->textEdit_Console->append(QString::fromStdString(e.getText()));
}

void MainWindow::_traceErrorHandler(TraceErrorEvent e) {
    ui->textEdit_Console->setTextColor(QColor::fromRgb(255, 0, 0));
    ui->textEdit_Console->append(QString::fromStdString(e.getText()));
}

void MainWindow::_traceReportHandler(TraceEvent e) {
    ui->textEdit_Console->setTextColor(QColor::fromRgb(0, 0, 128));
    ui->textEdit_Console->append(QString::fromStdString(e.getText()));
}

void MainWindow::_traceSimulationHandler(TraceSimulationEvent e) {
    if (e.getTracelevel() == Util::TraceLevel::L1_errorFatal)
        ui->textEdit_Console->setTextColor(QColor::fromRgb(255, 64, 0));
    else if (e.getTracelevel() == Util::TraceLevel::L3_errorRecover)
        ui->textEdit_Console->setTextColor(QColor::fromRgb(128, 64, 0));
    else {
        unsigned short grayVal = 5 * (static_cast<unsigned int> (e.getTracelevel()));
        ui->textEdit_Console->setTextColor(QColor::fromRgb(grayVal, 128, grayVal));
    }
    ui->textEdit_Console->append(QString::fromStdString(e.getText()));
}
// default Event Handlers

void MainWindow::_onSimulationStartHandler(SimulationEvent* re) {

}

void MainWindow::_onReplicationStartHandler(SimulationEvent* re) {

}

void MainWindow::_onProcessEventHandler(SimulationEvent* re) {

}

void MainWindow::_onReplicationEndHandler(SimulationEvent* re) {

}

void MainWindow::_onSimulationEndHandler(SimulationEvent* re) {

}

void MainWindow::_onEntityRemoveHandler(SimulationEvent* re) {

}

void MainWindow::_refreshActions() {
    bool thereIsAtLeastOneModelOpenned = simulator->getModels()->size() > 0;
    ui->actionClose->setEnabled(thereIsAtLeastOneModelOpenned);
    ui->actionSave->setEnabled(thereIsAtLeastOneModelOpenned);
    ui->actionSave_as->setEnabled(thereIsAtLeastOneModelOpenned);
    //ui->actionModels->setEnabled(thereIsAtLeastOneModelOpenned);
    ui->menuModel->setEnabled(thereIsAtLeastOneModelOpenned);
    ui->tabWidgetModels->setEnabled(thereIsAtLeastOneModelOpenned);
}

void MainWindow::_refreshTabWidgetModel() {
    bool thereIsAtLeastOneModelOpenned = simulator->getModels()->size() > 0;
    if (!thereIsAtLeastOneModelOpenned)
        ui->tabWidgetModels->clear(); // no models openned
}

//*****************************************************************
// MAINWINDOW START
//*****************************************************************

void MainWindow::on_actionAbout_triggered() {
    DialogAbout da;
    da.setLabelVersion(QString::fromStdString(this->simulator->getVersion()));
    da.setModal(true);
    da.exec();
}

void MainWindow::on_actionExit_triggered() {
    QMessageBox::StandardButton reply;
    reply = QMessageBox::question(this, "Exit", "Exit GeneSyS?", QMessageBox::Yes | QMessageBox::No);
    if (reply == QMessageBox::Yes) {
        //qDebug() « "Yes was clicked";
        QApplication::quit();
    } else {
        //qDebug() « "Yes was *not* clicked";
    }
```

```
}

void MainWindow::_createNewTabForModel(Model* newModel) {
    QMessageBox::StandardButton reply;
    reply = QMessageBox::question(this, "New", "Create new model?", QMessageBox::Yes |
 QMessageBox::No);
    if (reply == QMessageBox::Yes) {
        QWidget* newTabForModel = new QWidget(this);
        mapSimUI.map(newModel, newTabForModel);
        ui->tabWidgetModels->addTab(newTabForModel,
 QString::fromStdString(newModel->getInfos()->getName()));
        ui->tabWidgetModels->setCurrentIndex(ui->tabWidgetModels->count() - 1);
        _refreshActions();
        _refreshTabWidgetModel();
    }
}

void MainWindow::on_actionNew_triggered() {
    Model* newModel = new Model(simulator);
    simulator->getModels()->insert(newModel);
    _createNewTabForModel(newModel);
}

void MainWindow::on_actionOpen_triggered() {
    QString fileName = QFileDialog::getOpenFileName(this,
            tr("Open Simulation Model"), "",
            tr("Genesys Simulation Model (*.gsm);;All Files (*)"));
    if (fileName.isEmpty())
        return;
    else {
        QFile file(fileName);
        if (!file.open(QIODevice::ReadOnly)) {
            QMessageBox::information(this, tr("Unable to open file"),
                    file.errorString());
            return;
        }
        bool loaded = simulator->getModels()->loadModel(fileName.toStdString());
        if (loaded) {
            _createNewTabForModel(simulator->getModels()->current());
            _refreshActions();
            _refreshTabWidgetModel();
        }
    }
}

void MainWindow::on_actionClose_triggered() {

}

void MainWindow::on_actionSave_triggered() {
    QString fileName = QFileDialog::getSaveFileName(this,
            tr("Save Simulation Model"), "",
            tr("Genesys Simulation Model (*.gsm);;All Files (*)"));
    if (fileName.isEmpty())
        return;
    else {
        QFile file(fileName);
        if (!file.open(QIODevice::WriteOnly)) {
            QMessageBox::information(this, tr("Unable to open file"),
                    file.errorString());
            return;
        }
        //QDataStream out(&file);
        //out.setVersion(QDataStream::Qt_4_5);
        //out « contacts;
    }
}

void MainWindow::on_actionSave_as_triggered() {

}

void MainWindow::on_actionUndo_triggered() {

}

void MainWindow::on_actionRedo_triggered() {

}

void MainWindow::on_actionCut_triggered() {

}

void MainWindow::on_actionCopy_triggered() {

}
```

```
00369
00370 void MainWindow::on_actionPaste_triggered() {
00371
00372 }
00373
00374 void MainWindow::on_actionFind_triggered() {
00375
00376 }
00377
00378 void MainWindow::on_actionReplace_triggered() {
00379
00380 }
00381
00382 void MainWindow::on_actionGroup_triggered() {
00383
00384 }
00385
00386 void MainWindow::on_actionCheck_Model_triggered() {
00387
00388 }
00389
00390 void MainWindow::on_actionStart_triggered() {
00391
00392 }
00393
00394 void MainWindow::on_actionStep_triggered() {
00395
00396 }
00397
00398 void MainWindow::on_actionStop_triggered() {
00399
00400 }
00401
00402 void MainWindow::on_actionRun_Control_triggered() {
00403
00404 }
00405
00406 void MainWindow::on_tabWidgetModels_tabBarDoubleClicked(int index) {
00407
00408 }
00409
00410 void MainWindow::on_tabWidgetModels_tabCloseRequested(int index) {
00411
00412 }
00413
00414 void MainWindow::on_tabWidgetModels_currentChanged(int index) {
00415
00416 }
00417
00418 void MainWindow::on_actionInformation_triggered() {
00419     DialogModelInformation dmi;
00420     dmi.setModal(true);
00421     dmi.exec();
00422 }
```

8.15 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/main↩ Window/mainwindow.h File Reference

```
#include <QMainWindow>
#include "../mapkernelandui.h"
#include "../../../../kernel/simulator/Simulator.h"
#include "../../../../kernel/simulator/Plugin.h"
```

Include dependency graph for mainwindow.h:

This graph shows which files directly or indirectly include this file:

Classes

- class MainWindow

Namespaces

- Ui

8.16 mainwindow.h

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
#include "../mapkernelandui.h"
#include "../../../../kernel/simulator/Simulator.h"
#include "../../../../kernel/simulator/Plugin.h"
```

```
00008
00009 namespace Ui {
00010     class MainWindow;
00011 }
00012
00013 class MainWindow : public QMainWindow {
00014     Q_OBJECT
00015
00016 public:
00017     explicit MainWindow(QWidget *parent = 0);
00018     ~MainWindow();
00019
00020 private slots:
00021     void on_actionAbout_triggered();
00022
00023     void on_actionExit_triggered();
00024
00025     void on_actionNew_triggered();
00026
00027     void on_actionOpen_triggered();
00028
00029     void on_actionClose_triggered();
00030
00031     void on_actionSave_triggered();
00032
00033     void on_actionSave_as_triggered();
00034
00035     void on_actionUndo_triggered();
00036
00037     void on_actionRedo_triggered();
00038
00039     void on_actionCut_triggered();
00040
00041     void on_actionCopy_triggered();
00042
00043     void on_actionPaste_triggered();
00044
00045     void on_actionFind_triggered();
00046
00047     void on_actionReplace_triggered();
00048
00049     void on_actionGroup_triggered();
00050
00051     void on_actionCheck_Model_triggered();
00052
00053     void on_actionStart_triggered();
00054
00055     void on_actionStep_triggered();
00056
00057     void on_actionStop_triggered();
00058
00059     void on_actionRun_Control_triggered();
00060
00061     void on_tabWidgetModels_tabBarDoubleClicked(int index);
00062
00063     void on_tabWidgetModels_tabCloseRequested(int index);
00064
00065     void on_tabWidgetModels_currentChanged(int index);
00066
00067     void on_actionInformation_triggered();
00068
00069 private:
00070     Ui::MainWindow *ui;
00071     // dialogs
00072
00073
00074     //*****************************************
00075     // SIMULATOR
00076     //*****************************************
00077 private:
00078     void _refreshActions();
00079     void _refreshTabWidgetModel();
00080     void _createNewTabForModel(Model* newModel);
00081 private:
00082     void _traceHandler(TraceEvent e);
00083     void _traceErrorHandler(TraceErrorEvent e);
00084     void _traceReportHandler(TraceEvent e);
00085     void _traceSimulationHandler(TraceSimulationEvent e);
00086     // default Event Handlers
00087     void _onSimulationStartHandler(SimulationEvent* re);
00088     void _onReplicationStartHandler(SimulationEvent* re);
00089     void _onProcessEventHandler(SimulationEvent* re);
00090     void _onReplicationEndHandler(SimulationEvent* re);
00091     void _onSimulationEndHandler(SimulationEvent* re);
00092     void _onEntityRemoveHandler(SimulationEvent* re);
00093 private:
00094     void _simulationInsertPlugins();
```

```
    void __simulationInsert_FAKE_Plugins();
    void _insertPluginUI(Plugin* plugin);
private:
    Simulator* simulator;
    MapKernelAndUI mapSimUI;
};

#endif // MAINWINDOW_H
```

8.17 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mainWindow/ui_mainwindow.h File Reference

```
#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDockWidget>
#include <QtWidgets/QGraphicsView>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QListWidget>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QMenu>
#include <QtWidgets/QMenuBar>
#include <QtWidgets/QStatusBar>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QTableView>
#include <QtWidgets/QTextEdit>
#include <QtWidgets/QToolBar>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
```

This graph shows which files directly or indirectly include this file:

Classes

- class Ui_MainWindow
- class Ui::MainWindow

Namespaces

- Ui

8.18 ui_mainwindow.h

```
/********************************************************************************
 ** Form generated from reading UI file 'mainwindow.ui'
 **
 ** Created by: Qt User Interface Compiler version 5.15.2
 **
 ** WARNING! All changes made in this file will be lost when recompiling UI file!
 ********************************************************************************/

#ifndef UI_MAINWINDOW_H
#define UI_MAINWINDOW_H

#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDockWidget>
#include <QtWidgets/QGraphicsView>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QListWidget>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QMenu>
#include <QtWidgets/QMenuBar>
#include <QtWidgets/QStatusBar>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QTableView>
#include <QtWidgets/QTextEdit>
#include <QtWidgets/QToolBar>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>

QT_BEGIN_NAMESPACE

class Ui_MainWindow {
public:
    QAction *actionNew;
    QAction *actionOpen;
    QAction *actionClose;
    QAction *actionSave;
    QAction *actionSave_as;
    QAction *actionExit;
    QAction *actionUndo;
    QAction *actionRedo;
    QAction *actionCut;
    QAction *actionCopy;
    QAction *actionPaste;
    QAction *actionFind;
    QAction *actionReplace;
    QAction *actionGroup;
    QAction *actionZoom_In;
    QAction *actionZoom_Out;
    QAction *actionZoom_Factor;
    QAction *actionGrid;
    QAction *actionSnap;
    QAction *actionInput_Analyser;
    QAction *actionOutput_Analyser;
    QAction *actionScenario_Manager;
    QAction *actionOptimizaton;
    QAction *actionPreferences;
    QAction *actionCheck_Model;
    QAction *actionStart;
    QAction *actionStep;
    QAction *actionStop;
    QAction *actionRun_Control;
    QAction *actionComponents;
    QAction *actionElements;
    QAction *actionModels;
```

```
    QAction *actionConsole;
    QAction *actionProperties;
    QAction *actionEditor;
    QAction *actionDocumentation;
    QAction *actionSupport;
    QAction *actionAbout;
    QAction *actionTop;
    QAction *actionBottom;
    QAction *actionInformation;
    QAction *actionReport;
    QAction *actionComponents_2;
    QAction *actionElements_2;
    QAction *actionTrace;
    QAction *actionEvents;
    QAction *actionPause;
    QAction *actionExit_2;
    QAction *actionPreferences_2;
    QAction *actionModels_2;
    QAction *actionExperiments;
    QAction *actionConnect;
    QAction *actionZoom_In_2;
    QAction *actionZoom_Out_2;
    QAction *actionZoom_Facot;
    QAction *actionGrid_2;
    QAction *actionSnap_2;
    QAction *actionArranje;
    QAction *actionGroup_2;
    QAction *actionPlugins_2;
    QAction *actionInformation_2;
    QAction *actionSimulation;
    QAction *actionComponents_3;
    QAction *actionElements_3;
    QAction *actionLoad;
    QAction *actionSave_2;
    QAction *actionEvents_Diagram;
    QAction *actionArranje_2;
    QWidget *centralWidget;
    QGridLayout *gridLayout;
    QTabWidget *tabWidgetModels;
    QWidget *tabModel1;
    QHBoxLayout *horizontalLayout;
    QGraphicsView *graphicsView;
    QWidget *tabModel2;
    QHBoxLayout *horizontalLayout_2;
    QMenuBar *menuBar;
    QMenu *menuEdit;
    QMenu *menuModel;
    QMenu *menuView_2;
    QMenu *menuWindow;
    QMenu *menuHelp;
    QMenu *menuSimulator;
    QStatusBar *statusBar;
    QToolBar *toolBar;
    QDockWidget *dockWidgetPlugins;
    QWidget *dockWidgetPlugins_2;
    QVBoxLayout *verticalLayout;
    QListWidget *listWidget_Plugins;
    QDockWidget *dockWidgetProperties;
    QWidget *dockWidgetContents_Properties;
    QVBoxLayout *verticalLayout_3;
    QTableView *tableView_Properties;
    QDockWidget *dockWidgetConsole;
    QWidget *dockWidgetContents_Console;
    QVBoxLayout *verticalLayout_4;
    QTextEdit *textEdit_Console;
    QDockWidget *dockWidgetReports;
    QWidget *dockWidgetContents_Reports;
    QVBoxLayout *verticalLayout_5;
    QTextEdit *textEdit_Report;
    QDockWidget *dockWidgetExperiments;
    QWidget *dockWidgetContents_Experiments;
    QListWidget *listWidget_Experiments;
    void setupUi(QMainWindow *MainWindow) {
        if (MainWindow->objectName().isEmpty())
            MainWindow->setObjectName(QString::fromUtf8("MainWindow"));
        MainWindow->resize(1273, 724);
        MainWindow->setLocale(QLocale(QLocale::English, QLocale::UnitedStates));
        actionNew = new QAction(MainWindow);
        actionNew->setObjectName(QString::fromUtf8("actionNew"));
        QIcon icon;
        icon.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/60.png"), QSize(),
QIcon::Normal, QIcon::Off);
        actionNew->setIcon(icon);
        actionOpen = new QAction(MainWindow);
        actionOpen->setObjectName(QString::fromUtf8("actionOpen"));
        QIcon icon1;
        icon1.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/52.png"), QSize(),
```

```
QIcon::Normal, QIcon::Off);
        actionOpen->setIcon(icon1);
        actionClose = new QAction(MainWindow);
        actionClose->setObjectName(QString::fromUtf8("actionClose"));
        actionClose->setEnabled(true);
        QIcon icon2;
        icon2.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Close.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionClose->setIcon(icon2);
        actionSave = new QAction(MainWindow);
        actionSave->setObjectName(QString::fromUtf8("actionSave"));
        actionSave->setEnabled(true);
        QIcon icon3;
        icon3.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Save.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionSave->setIcon(icon3);
        actionSave_as = new QAction(MainWindow);
        actionSave_as->setObjectName(QString::fromUtf8("actionSave_as"));
        actionSave_as->setEnabled(true);
        QIcon icon4;
        icon4.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/86.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionSave_as->setIcon(icon4);
        actionExit = new QAction(MainWindow);
        actionExit->setObjectName(QString::fromUtf8("actionExit"));
        QIcon icon5;
        icon5.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Door.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionExit->setIcon(icon5);
        actionUndo = new QAction(MainWindow);
        actionUndo->setObjectName(QString::fromUtf8("actionUndo"));
        actionUndo->setEnabled(false);
        QIcon icon6;
        icon6.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Undo.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionUndo->setIcon(icon6);
        actionRedo = new QAction(MainWindow);
        actionRedo->setObjectName(QString::fromUtf8("actionRedo"));
        actionRedo->setEnabled(false);
        QIcon icon7;
        icon7.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Redo.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionRedo->setIcon(icon7);
        actionCut = new QAction(MainWindow);
        actionCut->setObjectName(QString::fromUtf8("actionCut"));
        actionCut->setEnabled(false);
        QIcon icon8;
        icon8.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Cut.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionCut->setIcon(icon8);
        actionCopy = new QAction(MainWindow);
        actionCopy->setObjectName(QString::fromUtf8("actionCopy"));
        actionCopy->setEnabled(false);
        QIcon icon9;
        icon9.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Copy.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionCopy->setIcon(icon9);
        actionPaste = new QAction(MainWindow);
        actionPaste->setObjectName(QString::fromUtf8("actionPaste"));
        actionPaste->setEnabled(false);
        QIcon icon10;
        icon10.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Paste.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionPaste->setIcon(icon10);
        actionFind = new QAction(MainWindow);
        actionFind->setObjectName(QString::fromUtf8("actionFind"));
        QIcon icon11;
        icon11.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Find.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionFind->setIcon(icon11);
        actionReplace = new QAction(MainWindow);
        actionReplace->setObjectName(QString::fromUtf8("actionReplace"));
        QIcon icon12;
        icon12.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/View.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionReplace->setIcon(icon12);
        actionGroup = new QAction(MainWindow);
        actionGroup->setObjectName(QString::fromUtf8("actionGroup"));
        actionZoom_In = new QAction(MainWindow);
        actionZoom_In->setObjectName(QString::fromUtf8("actionZoom_In"));
        QIcon icon13;
        icon13.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Zoom.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionZoom_In->setIcon(icon13);
        actionZoom_Out = new QAction(MainWindow);
        actionZoom_Out->setObjectName(QString::fromUtf8("actionZoom_Out"));
        actionZoom_Out->setIcon(icon13);
```

```
        actionZoom_Factor = new QAction(MainWindow);
        actionZoom_Factor->setObjectName(QString::fromUtf8("actionZoom_Factor"));
        actionZoom_Factor->setIcon(icon13);
        actionGrid = new QAction(MainWindow);
        actionGrid->setObjectName(QString::fromUtf8("actionGrid"));
        actionSnap = new QAction(MainWindow);
        actionSnap->setObjectName(QString::fromUtf8("actionSnap"));
        actionInput_Analyser = new QAction(MainWindow);
        actionInput_Analyser->setObjectName(QString::fromUtf8("actionInput_Analyser"));
        actionOutput_Analyser = new QAction(MainWindow);
        actionOutput_Analyser->setObjectName(QString::fromUtf8("actionOutput_Analyser"));
        actionScenario_Manager = new QAction(MainWindow);
        actionScenario_Manager->setObjectName(QString::fromUtf8("actionScenario_Manager"));
        actionOptimizaton = new QAction(MainWindow);
        actionOptimizaton->setObjectName(QString::fromUtf8("actionOptimizaton"));
        actionPreferences = new QAction(MainWindow);
        actionPreferences->setObjectName(QString::fromUtf8("actionPreferences"));
        actionCheck_Model = new QAction(MainWindow);
        actionCheck_Model->setObjectName(QString::fromUtf8("actionCheck_Model"));
        QIcon icon14;
        icon14.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/58.png"), QSize(),
    QIcon::Normal, QIcon::Off);
        actionCheck_Model->setIcon(icon14);
        actionStart = new QAction(MainWindow);
        actionStart->setObjectName(QString::fromUtf8("actionStart"));
        QIcon icon15;
        icon15.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Play.png"), QSize(),
    QIcon::Normal, QIcon::Off);
        actionStart->setIcon(icon15);
        actionStep = new QAction(MainWindow);
        actionStep->setObjectName(QString::fromUtf8("actionStep"));
        QIcon icon16;
        icon16.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Next track.png"),
    QSize(), QIcon::Normal, QIcon::Off);
        actionStep->setIcon(icon16);
        actionStop = new QAction(MainWindow);
        actionStop->setObjectName(QString::fromUtf8("actionStop"));
        QIcon icon17;
        icon17.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Stop.png"), QSize(),
    QIcon::Normal, QIcon::Off);
        actionStop->setIcon(icon17);
        actionRun_Control = new QAction(MainWindow);
        actionRun_Control->setObjectName(QString::fromUtf8("actionRun_Control"));
        QIcon icon18;
        icon18.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Tune.png"), QSize(),
    QIcon::Normal, QIcon::Off);
        actionRun_Control->setIcon(icon18);
        actionComponents = new QAction(MainWindow);
        actionComponents->setObjectName(QString::fromUtf8("actionComponents"));
        actionElements = new QAction(MainWindow);
        actionElements->setObjectName(QString::fromUtf8("actionElements"));
        actionModels = new QAction(MainWindow);
        actionModels->setObjectName(QString::fromUtf8("actionModels"));
        actionConsole = new QAction(MainWindow);
        actionConsole->setObjectName(QString::fromUtf8("actionConsole"));
        actionProperties = new QAction(MainWindow);
        actionProperties->setObjectName(QString::fromUtf8("actionProperties"));
        actionEditor = new QAction(MainWindow);
        actionEditor->setObjectName(QString::fromUtf8("actionEditor"));
        actionDocumentation = new QAction(MainWindow);
        actionDocumentation->setObjectName(QString::fromUtf8("actionDocumentation"));
        QIcon icon19;
        icon19.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Help book.png"),
    QSize(), QIcon::Normal, QIcon::Off);
        actionDocumentation->setIcon(icon19);
        actionSupport = new QAction(MainWindow);
        actionSupport->setObjectName(QString::fromUtf8("actionSupport"));
        QIcon icon20;
        icon20.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Help.png"), QSize(),
    QIcon::Normal, QIcon::Off);
        actionSupport->setIcon(icon20);
        actionAbout = new QAction(MainWindow);
        actionAbout->setObjectName(QString::fromUtf8("actionAbout"));
        QIcon icon21;
        icon21.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Help book 3d.png"),
    QSize(), QIcon::Normal, QIcon::Off);
        actionAbout->setIcon(icon21);
        actionTop = new QAction(MainWindow);
        actionTop->setObjectName(QString::fromUtf8("actionTop"));
        actionBottom = new QAction(MainWindow);
        actionBottom->setObjectName(QString::fromUtf8("actionBottom"));
        actionInformation = new QAction(MainWindow);
        actionInformation->setObjectName(QString::fromUtf8("actionInformation"));
        actionInformation->setEnabled(true);
        actionReport = new QAction(MainWindow);
        actionReport->setObjectName(QString::fromUtf8("actionReport"));
        actionComponents_2 = new QAction(MainWindow);
```

```
        actionComponents_2->setObjectName(QString::fromUtf8("actionComponents_2"));
        actionElements_2 = new QAction(MainWindow);
        actionElements_2->setObjectName(QString::fromUtf8("actionElements_2"));
        actionTrace = new QAction(MainWindow);
        actionTrace->setObjectName(QString::fromUtf8("actionTrace"));
        actionEvents = new QAction(MainWindow);
        actionEvents->setObjectName(QString::fromUtf8("actionEvents"));
        actionPause = new QAction(MainWindow);
        actionPause->setObjectName(QString::fromUtf8("actionPause"));
        QIcon icon22;
        icon22.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Pause.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionPause->setIcon(icon22);
        actionExit_2 = new QAction(MainWindow);
        actionExit_2->setObjectName(QString::fromUtf8("actionExit_2"));
        actionPreferences_2 = new QAction(MainWindow);
        actionPreferences_2->setObjectName(QString::fromUtf8("actionPreferences_2"));
        actionModels_2 = new QAction(MainWindow);
        actionModels_2->setObjectName(QString::fromUtf8("actionModels_2"));
        actionExperiments = new QAction(MainWindow);
        actionExperiments->setObjectName(QString::fromUtf8("actionExperiments"));
        actionConnect = new QAction(MainWindow);
        actionConnect->setObjectName(QString::fromUtf8("actionConnect"));
        actionZoom_In_2 = new QAction(MainWindow);
        actionZoom_In_2->setObjectName(QString::fromUtf8("actionZoom_In_2"));
        actionZoom_Out_2 = new QAction(MainWindow);
        actionZoom_Out_2->setObjectName(QString::fromUtf8("actionZoom_Out_2"));
        actionZoom_Facot = new QAction(MainWindow);
        actionZoom_Facot->setObjectName(QString::fromUtf8("actionZoom_Facot"));
        actionGrid_2 = new QAction(MainWindow);
        actionGrid_2->setObjectName(QString::fromUtf8("actionGrid_2"));
        actionSnap_2 = new QAction(MainWindow);
        actionSnap_2->setObjectName(QString::fromUtf8("actionSnap_2"));
        actionArranje = new QAction(MainWindow);
        actionArranje->setObjectName(QString::fromUtf8("actionArranje"));
        actionGroup_2 = new QAction(MainWindow);
        actionGroup_2->setObjectName(QString::fromUtf8("actionGroup_2"));
        actionPlugins_2 = new QAction(MainWindow);
        actionPlugins_2->setObjectName(QString::fromUtf8("actionPlugins_2"));
        actionInformation_2 = new QAction(MainWindow);
        actionInformation_2->setObjectName(QString::fromUtf8("actionInformation_2"));
        actionSimulation = new QAction(MainWindow);
        actionSimulation->setObjectName(QString::fromUtf8("actionSimulation"));
        actionComponents_3 = new QAction(MainWindow);
        actionComponents_3->setObjectName(QString::fromUtf8("actionComponents_3"));
        actionElements_3 = new QAction(MainWindow);
        actionElements_3->setObjectName(QString::fromUtf8("actionElements_3"));
        actionLoad = new QAction(MainWindow);
        actionLoad->setObjectName(QString::fromUtf8("actionLoad"));
        actionSave_2 = new QAction(MainWindow);
        actionSave_2->setObjectName(QString::fromUtf8("actionSave_2"));
        actionEvents_Diagram = new QAction(MainWindow);
        actionEvents_Diagram->setObjectName(QString::fromUtf8("actionEvents_Diagram"));
        actionArranje_2 = new QAction(MainWindow);
        actionArranje_2->setObjectName(QString::fromUtf8("actionArranje_2"));
        centralWidget = new QWidget(MainWindow);
        centralWidget->setObjectName(QString::fromUtf8("centralWidget"));
        gridLayout = new QGridLayout(centralWidget);
        gridLayout->setSpacing(6);
        gridLayout->setContentsMargins(11, 11, 11, 11);
        gridLayout->setObjectName(QString::fromUtf8("gridLayout"));
        tabWidgetModels = new QTabWidget(centralWidget);
        tabWidgetModels->setObjectName(QString::fromUtf8("tabWidgetModels"));
        tabWidgetModels->setTabPosition(QTabWidget::North);
        tabWidgetModels->setTabShape(QTabWidget::Rounded);
        tabWidgetModels->setTabsClosable(true);
        tabWidgetModels->setMovable(true);
        tabModel1 = new QWidget();
        tabModel1->setObjectName(QString::fromUtf8("tabModel1"));
        horizontalLayout = new QHBoxLayout(tabModel1);
        horizontalLayout->setSpacing(6);
        horizontalLayout->setContentsMargins(11, 11, 11, 11);
        horizontalLayout->setObjectName(QString::fromUtf8("horizontalLayout"));
        graphicsView = new QGraphicsView(tabModel1);
        graphicsView->setObjectName(QString::fromUtf8("graphicsView"));
        graphicsView->setFrameShape(QFrame::WinPanel);
        graphicsView->setFrameShadow(QFrame::Sunken);
        graphicsView->setLineWidth(2);
        graphicsView->setMidLineWidth(1);
        graphicsView->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        graphicsView->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        QBrush brush(QColor(238, 238, 238, 255));
        brush.setStyle(Qt::SolidPattern);
        graphicsView->setBackgroundBrush(brush);
        graphicsView->setDragMode(QGraphicsView::NoDrag);

        horizontalLayout->addWidget(graphicsView);
```

```

        tabWidgetModels->addTab(tabModel1, QString());
        tabModel2 = new QWidget();
        tabModel2->setObjectName(QString::fromUtf8("tabModel2"));
        horizontalLayout_2 = new QHBoxLayout(tabModel2);
        horizontalLayout_2->setSpacing(6);
        horizontalLayout_2->setContentsMargins(11, 11, 11, 11);
        horizontalLayout_2->setObjectName(QString::fromUtf8("horizontalLayout_2"));
        tabWidgetModels->addTab(tabModel2, QString());

        gridLayout->addWidget(tabWidgetModels, 0, 0, 1, 1);

        MainWindow->setCentralWidget(centralWidget);
        menuBar = new QMenuBar(MainWindow);
        menuBar->setObjectName(QString::fromUtf8("menuBar"));
        menuBar->setGeometry(QRect(0, 0, 1273, 24));
        menuEdit = new QMenu(menuBar);
        menuEdit->setObjectName(QString::fromUtf8("menuEdit"));
        menuModel = new QMenu(menuBar);
        menuModel->setObjectName(QString::fromUtf8("menuModel"));
        menuModel->setEnabled(true);
        menuView_2 = new QMenu(menuModel);
        menuView_2->setObjectName(QString::fromUtf8("menuView_2"));
        menuWindow = new QMenu(menuBar);
        menuWindow->setObjectName(QString::fromUtf8("menuWindow"));
        menuHelp = new QMenu(menuBar);
        menuHelp->setObjectName(QString::fromUtf8("menuHelp"));
        menuSimulator = new QMenu(menuBar);
        menuSimulator->setObjectName(QString::fromUtf8("menuSimulator"));
        MainWindow->setMenuBar(menuBar);
        statusBar = new QStatusBar(MainWindow);
        statusBar->setObjectName(QString::fromUtf8("statusBar"));
        MainWindow->setStatusBar(statusBar);
        toolBar = new QToolBar(MainWindow);
        toolBar->setObjectName(QString::fromUtf8("toolBar"));
        MainWindow->addToolBar(Qt::TopToolBarArea, toolBar);
        dockWidgetPlugins = new QDockWidget(MainWindow);
        dockWidgetPlugins->setObjectName(QString::fromUtf8("dockWidgetPlugins"));
        dockWidgetPlugins->setAutoFillBackground(false);
        dockWidgetPlugins->setFeatures(QDockWidget::DockWidgetFloatable |
QDockWidget::DockWidgetMovable);
        dockWidgetPlugins_2 = new QWidget();
        dockWidgetPlugins_2->setObjectName(QString::fromUtf8("dockWidgetPlugins_2"));
        verticalLayout = new QVBoxLayout(dockWidgetPlugins_2);
        verticalLayout->setSpacing(6);
        verticalLayout->setContentsMargins(11, 11, 11, 11);
        verticalLayout->setObjectName(QString::fromUtf8("verticalLayout"));
        listWidget_Plugins = new QListWidget(dockWidgetPlugins_2);
        listWidget_Plugins->setObjectName(QString::fromUtf8("listWidget_Plugins"));

        verticalLayout->addWidget(listWidget_Plugins);

        dockWidgetPlugins->setWidget(dockWidgetPlugins_2);
        MainWindow->addDockWidget(Qt::LeftDockWidgetArea, dockWidgetPlugins);
        dockWidgetProperties = new QDockWidget(MainWindow);
        dockWidgetProperties->setObjectName(QString::fromUtf8("dockWidgetProperties"));
        dockWidgetProperties->setFeatures(QDockWidget::DockWidgetFloatable |
QDockWidget::DockWidgetMovable);
        dockWidgetContents_Properties = new QWidget();

dockWidgetContents_Properties->setObjectName(QString::fromUtf8("dockWidgetContents_Properties"));
        verticalLayout_3 = new QVBoxLayout(dockWidgetContents_Properties);
        verticalLayout_3->setSpacing(6);
        verticalLayout_3->setContentsMargins(11, 11, 11, 11);
        verticalLayout_3->setObjectName(QString::fromUtf8("verticalLayout_3"));
        tableView_Properties = new QTableView(dockWidgetContents_Properties);
        tableView_Properties->setObjectName(QString::fromUtf8("tableView_Properties"));
        tableView_Properties->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        tableView_Properties->setHorizontalScrollBarPolicy(Qt::ScrollBarAsNeeded);

        verticalLayout_3->addWidget(tableView_Properties);

        dockWidgetProperties->setWidget(dockWidgetContents_Properties);
        MainWindow->addDockWidget(Qt::RightDockWidgetArea, dockWidgetProperties);
        dockWidgetConsole = new QDockWidget(MainWindow);
        dockWidgetConsole->setObjectName(QString::fromUtf8("dockWidgetConsole"));
        dockWidgetConsole->setFloating(false);
        dockWidgetConsole->setFeatures(QDockWidget::DockWidgetFloatable |
QDockWidget::DockWidgetMovable);
        dockWidgetConsole->setAllowedAreas(Qt::AllDockWidgetAreas);
        dockWidgetContents_Console = new QWidget();
        dockWidgetContents_Console->setObjectName(QString::fromUtf8("dockWidgetContents_Console"));
        verticalLayout_4 = new QVBoxLayout(dockWidgetContents_Console);
        verticalLayout_4->setSpacing(6);
        verticalLayout_4->setContentsMargins(11, 11, 11, 11);
        verticalLayout_4->setObjectName(QString::fromUtf8("verticalLayout_4"));
        textEdit_Console = new QTextEdit(dockWidgetContents_Console);
```

```
        textEdit_Console->setObjectName(QString::fromUtf8("textEdit_Console"));
        QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
        sizePolicy.setHorizontalStretch(0);
        sizePolicy.setVerticalStretch(0);
        sizePolicy.setHeightForWidth(textEdit_Console->sizePolicy().hasHeightForWidth());
        textEdit_Console->setSizePolicy(sizePolicy);
        textEdit_Console->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        textEdit_Console->setLineWrapMode(QTextEdit::NoWrap);
        textEdit_Console->setReadOnly(true);
        textEdit_Console->setTextInteractionFlags(Qt::NoTextInteraction);

        verticalLayout_4->addWidget(textEdit_Console);

        dockWidgetConsole->setWidget(dockWidgetContents_Console);
        MainWindow->addDockWidget(Qt::BottomDockWidgetArea, dockWidgetConsole);
        dockWidgetReports = new QDockWidget(MainWindow);
        dockWidgetReports->setObjectName(QString::fromUtf8("dockWidgetReports"));
        dockWidgetReports->setFeatures(QDockWidget::DockWidgetFloatable |
QDockWidget::DockWidgetMovable);
        dockWidgetContents_Reports = new QWidget();
        dockWidgetContents_Reports->setObjectName(QString::fromUtf8("dockWidgetContents_Reports"));
        verticalLayout_5 = new QVBoxLayout(dockWidgetContents_Reports);
        verticalLayout_5->setSpacing(6);
        verticalLayout_5->setContentsMargins(11, 11, 11, 11);
        verticalLayout_5->setObjectName(QString::fromUtf8("verticalLayout_5"));
        textEdit_Report = new QTextEdit(dockWidgetContents_Reports);
        textEdit_Report->setObjectName(QString::fromUtf8("textEdit_Report"));
        sizePolicy.setHeightForWidth(textEdit_Report->sizePolicy().hasHeightForWidth());
        textEdit_Report->setSizePolicy(sizePolicy);
        textEdit_Report->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);

        verticalLayout_5->addWidget(textEdit_Report);

        dockWidgetReports->setWidget(dockWidgetContents_Reports);
        MainWindow->addDockWidget(Qt::BottomDockWidgetArea, dockWidgetReports);
        dockWidgetExperiments = new QDockWidget(MainWindow);
        dockWidgetExperiments->setObjectName(QString::fromUtf8("dockWidgetExperiments"));
        dockWidgetContents_Experiments = new QWidget();

dockWidgetContents_Experiments->setObjectName(QString::fromUtf8("dockWidgetContents_Experiments"));
        dockWidgetContents_Experiments->setMinimumSize(QSize(0, 144));
        listWidget_Experiments = new QListWidget(dockWidgetContents_Experiments);
        listWidget_Experiments->setObjectName(QString::fromUtf8("listWidget_Experiments"));
        listWidget_Experiments->setGeometry(QRect(10, 10, 201, 161));
        listWidget_Experiments->setAutoFillBackground(false);
        dockWidgetExperiments->setWidget(dockWidgetContents_Experiments);
        MainWindow->addDockWidget(Qt::LeftDockWidgetArea, dockWidgetExperiments);

        menuBar->addAction(menuSimulator->menuAction());
        menuBar->addAction(menuEdit->menuAction());
        menuBar->addAction(menuModel->menuAction());
        menuBar->addAction(menuWindow->menuAction());
        menuBar->addAction(menuHelp->menuAction());
        menuEdit->addAction(actionUndo);
        menuEdit->addAction(actionRedo);
        menuEdit->addSeparator();
        menuEdit->addAction(actionCut);
        menuEdit->addAction(actionCopy);
        menuEdit->addAction(actionPaste);
        menuEdit->addSeparator();
        menuEdit->addAction(actionFind);
        menuEdit->addAction(actionReplace);
        menuModel->addAction(actionInformation_2);
        menuModel->addAction(actionSimulation);
        menuModel->addSeparator();
        menuModel->addAction(actionComponents_3);
        menuModel->addAction(actionElements_3);
        menuModel->addSeparator();
        menuModel->addAction(actionSave_2);
        menuModel->addSeparator();
        menuModel->addAction(menuView_2->menuAction());
        menuView_2->addAction(actionZoom_In_2);
        menuView_2->addAction(actionZoom_Out_2);
        menuView_2->addAction(actionZoom_Facot);
        menuView_2->addSeparator();
        menuView_2->addAction(actionGrid_2);
        menuView_2->addAction(actionSnap_2);
        menuView_2->addSeparator();
        menuView_2->addAction(actionArranje);
        menuView_2->addAction(actionGroup_2);
        menuWindow->addAction(actionComponents);
        menuWindow->addAction(actionElements);
        menuWindow->addAction(actionProperties);
        menuWindow->addAction(actionEditor);
        menuWindow->addAction(actionConsole);
        menuWindow->addSeparator();
        menuWindow->addAction(actionModels);
```

```
        menuWindow->addSeparator();
        menuWindow->addAction(actionArranje_2);
        menuHelp->addAction(actionDocumentation);
        menuHelp->addAction(actionSupport);
        menuHelp->addSeparator();
        menuHelp->addAction(actionAbout);
        menuSimulator->addSeparator();
        menuSimulator->addAction(actionPlugins_2);
        menuSimulator->addAction(actionExperiments);
        menuSimulator->addAction(actionModels_2);
        menuSimulator->addSeparator();
        menuSimulator->addAction(actionPreferences_2);
        menuSimulator->addSeparator();
        menuSimulator->addAction(actionExit_2);
        toolBar->addAction(actionZoom_Out);
        toolBar->addAction(actionZoom_Factor);

        retranslateUi(MainWindow);

        tabWidgetModels->setCurrentIndex(0);


        QMetaObject::connectSlotsByName(MainWindow);
    } // setupUi
    void retranslateUi(QMainWindow *MainWindow) {
        MainWindow->setWindowTitle(QCoreApplication::translate("MainWindow", "GenESyS -- Generic and Expansible System Simulator", nullptr));
        actionNew->setText(QCoreApplication::translate("MainWindow", "New", nullptr));
#if QT_CONFIG(tooltip)
        actionNew->setToolTip(QCoreApplication::translate("MainWindow", "New simulation model", nullptr));
#endif // QT_CONFIG(tooltip)
        actionOpen->setText(QCoreApplication::translate("MainWindow", "Open", nullptr));
#if QT_CONFIG(shortcut)
        actionOpen->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+O", nullptr));
#endif // QT_CONFIG(shortcut)
        actionClose->setText(QCoreApplication::translate("MainWindow", "Close", nullptr));
#if QT_CONFIG(shortcut)
        actionClose->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+W", nullptr));
#endif // QT_CONFIG(shortcut)
        actionSave->setText(QCoreApplication::translate("MainWindow", "Save", nullptr));
#if QT_CONFIG(shortcut)
        actionSave->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+S", nullptr));
#endif // QT_CONFIG(shortcut)
        actionSave_as->setText(QCoreApplication::translate("MainWindow", "Save all", nullptr));
#if QT_CONFIG(shortcut)
        actionSave_as->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+A", nullptr));
#endif // QT_CONFIG(shortcut)
        actionExit->setText(QCoreApplication::translate("MainWindow", "Exit", nullptr));
#if QT_CONFIG(shortcut)
        actionExit->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+F4", nullptr));
#endif // QT_CONFIG(shortcut)
        actionUndo->setText(QCoreApplication::translate("MainWindow", "Undo", nullptr));
#if QT_CONFIG(shortcut)
        actionUndo->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+Z", nullptr));
#endif // QT_CONFIG(shortcut)
        actionRedo->setText(QCoreApplication::translate("MainWindow", "Redo", nullptr));
#if QT_CONFIG(shortcut)
        actionRedo->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+Shift+Z", nullptr));
#endif // QT_CONFIG(shortcut)
        actionCut->setText(QCoreApplication::translate("MainWindow", "Cut", nullptr));
#if QT_CONFIG(shortcut)
        actionCut->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+X", nullptr));
#endif // QT_CONFIG(shortcut)
        actionCopy->setText(QCoreApplication::translate("MainWindow", "Copy", nullptr));
#if QT_CONFIG(shortcut)
        actionCopy->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+C", nullptr));
#endif // QT_CONFIG(shortcut)
        actionPaste->setText(QCoreApplication::translate("MainWindow", "Paste", nullptr));
#if QT_CONFIG(shortcut)
        actionPaste->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+V", nullptr));
#endif // QT_CONFIG(shortcut)
        actionFind->setText(QCoreApplication::translate("MainWindow", "Find", nullptr));
#if QT_CONFIG(shortcut)
        actionFind->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+F", nullptr));
#endif // QT_CONFIG(shortcut)
        actionReplace->setText(QCoreApplication::translate("MainWindow", "Replace", nullptr));
#if QT_CONFIG(shortcut)
        actionReplace->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+H", nullptr));
#endif // QT_CONFIG(shortcut)
        actionGroup->setText(QCoreApplication::translate("MainWindow", "Group", nullptr));
        actionZoom_In->setText(QCoreApplication::translate("MainWindow", "Zoom In", nullptr));
#if QT_CONFIG(shortcut)
        actionZoom_In->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+=", nullptr));
#endif // QT_CONFIG(shortcut)
        actionZoom_Out->setText(QCoreApplication::translate("MainWindow", "Zoom Out", nullptr));
#if QT_CONFIG(shortcut)
```

```
        actionZoom_Out->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+-", nullptr));
#endif // QT_CONFIG(shortcut)
        actionZoom_Factor->setText(QCoreApplication::translate("MainWindow", "Zoom Factor", nullptr));
        actionGrid->setText(QCoreApplication::translate("MainWindow", "Grid", nullptr));
        actionSnap->setText(QCoreApplication::translate("MainWindow", "Snap", nullptr));
        actionInput_Analyser->setText(QCoreApplication::translate("MainWindow", "Input Analyser", nullptr));
        actionOutput_Analyser->setText(QCoreApplication::translate("MainWindow", "Output Analyser", nullptr));
        actionScenario_Manager->setText(QCoreApplication::translate("MainWindow", "Scenario Manager", nullptr));
        actionOptimizaton->setText(QCoreApplication::translate("MainWindow", "Optimizaton", nullptr));
        actionPreferences->setText(QCoreApplication::translate("MainWindow", "Preferences", nullptr));
        actionCheck_Model->setText(QCoreApplication::translate("MainWindow", "Check", nullptr));
#if QT_CONFIG(shortcut)
        actionCheck_Model->setShortcut(QCoreApplication::translate("MainWindow", "F4", nullptr));
#endif // QT_CONFIG(shortcut)
        actionStart->setText(QCoreApplication::translate("MainWindow", "Start", nullptr));
#if QT_CONFIG(shortcut)
        actionStart->setShortcut(QCoreApplication::translate("MainWindow", "F5", nullptr));
#endif // QT_CONFIG(shortcut)
        actionStep->setText(QCoreApplication::translate("MainWindow", "Step", nullptr));
#if QT_CONFIG(shortcut)
        actionStep->setShortcut(QCoreApplication::translate("MainWindow", "F7", nullptr));
#endif // QT_CONFIG(shortcut)
        actionStop->setText(QCoreApplication::translate("MainWindow", "Stop", nullptr));
#if QT_CONFIG(shortcut)
        actionStop->setShortcut(QCoreApplication::translate("MainWindow", "F10", nullptr));
#endif // QT_CONFIG(shortcut)
        actionRun_Control->setText(QCoreApplication::translate("MainWindow", "Settings", nullptr));
        actionComponents->setText(QCoreApplication::translate("MainWindow", "Plugins", nullptr));
        actionElements->setText(QCoreApplication::translate("MainWindow", "Experiments", nullptr));
        actionModels->setText(QCoreApplication::translate("MainWindow", "Models", nullptr));
        actionConsole->setText(QCoreApplication::translate("MainWindow", "Console", nullptr));
        actionProperties->setText(QCoreApplication::translate("MainWindow", "Properties", nullptr));
        actionEditor->setText(QCoreApplication::translate("MainWindow", "Reports", nullptr));
        actionDocumentation->setText(QCoreApplication::translate("MainWindow", "Documentation", nullptr));
#if QT_CONFIG(shortcut)
        actionDocumentation->setShortcut(QCoreApplication::translate("MainWindow", "F1", nullptr));
#endif // QT_CONFIG(shortcut)
        actionSupport->setText(QCoreApplication::translate("MainWindow", "Support", nullptr));
#if QT_CONFIG(shortcut)
        actionSupport->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+F1", nullptr));
#endif // QT_CONFIG(shortcut)
        actionAbout->setText(QCoreApplication::translate("MainWindow", "About...", nullptr));
#if QT_CONFIG(shortcut)
        actionAbout->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+Shift+F1", nullptr));
#endif // QT_CONFIG(shortcut)
        actionTop->setText(QCoreApplication::translate("MainWindow", "Top", nullptr));
        actionBottom->setText(QCoreApplication::translate("MainWindow", "Bottom", nullptr));
        actionInformation->setText(QCoreApplication::translate("MainWindow", "Information", nullptr));
        actionReport->setText(QCoreApplication::translate("MainWindow", "Report", nullptr));
        actionComponents_2->setText(QCoreApplication::translate("MainWindow", "Components", nullptr));
        actionElements_2->setText(QCoreApplication::translate("MainWindow", "Elements", nullptr));
        actionTrace->setText(QCoreApplication::translate("MainWindow", "Trace", nullptr));
        actionEvents->setText(QCoreApplication::translate("MainWindow", "Events", nullptr));
        actionPause->setText(QCoreApplication::translate("MainWindow", "Pause", nullptr));
        actionExit_2->setText(QCoreApplication::translate("MainWindow", "Exit", nullptr));
        actionPreferences_2->setText(QCoreApplication::translate("MainWindow", "Preferences", nullptr));
        actionModels_2->setText(QCoreApplication::translate("MainWindow", "Models", nullptr));
        actionExperiments->setText(QCoreApplication::translate("MainWindow", "Experiments", nullptr));
        actionConnect->setText(QCoreApplication::translate("MainWindow", "Connect", nullptr));
        actionZoom_In_2->setText(QCoreApplication::translate("MainWindow", "Zoom In", nullptr));
        actionZoom_Out_2->setText(QCoreApplication::translate("MainWindow", "Zoom Out", nullptr));
        actionZoom_Facot->setText(QCoreApplication::translate("MainWindow", "Zoom Facot", nullptr));
        actionGrid_2->setText(QCoreApplication::translate("MainWindow", "Grid", nullptr));
        actionSnap_2->setText(QCoreApplication::translate("MainWindow", "Snap", nullptr));
        actionArranje->setText(QCoreApplication::translate("MainWindow", "Arranje", nullptr));
        actionGroup_2->setText(QCoreApplication::translate("MainWindow", "Group", nullptr));
        actionPlugins_2->setText(QCoreApplication::translate("MainWindow", "Plugins", nullptr));
        actionInformation_2->setText(QCoreApplication::translate("MainWindow", "Information", nullptr));
        actionSimulation->setText(QCoreApplication::translate("MainWindow", "Simulation", nullptr));
        actionComponents_3->setText(QCoreApplication::translate("MainWindow", "Components", nullptr));
        actionElements_3->setText(QCoreApplication::translate("MainWindow", "Elements", nullptr));
        actionLoad->setText(QCoreApplication::translate("MainWindow", "Load", nullptr));
        actionSave_2->setText(QCoreApplication::translate("MainWindow", "Save", nullptr));
        actionEvents_Diagram->setText(QCoreApplication::translate("MainWindow", "Events Diagram", nullptr));
        actionArranje_2->setText(QCoreApplication::translate("MainWindow", "Arranje", nullptr));
        tabWidgetModels->setTabText(tabWidgetModels->indexOf(tabModel1), QCoreApplication::translate("MainWindow", "Model 1", nullptr));
        tabWidgetModels->setTabText(tabWidgetModels->indexOf(tabModel2), QCoreApplication::translate("MainWindow", "Model 2", nullptr));
        menuEdit->setTitle(QCoreApplication::translate("MainWindow", "Edit", nullptr));
```

```
        menuModel->setTitle(QCoreApplication::translate("MainWindow", "Model", nullptr));
        menuView_2->setTitle(QCoreApplication::translate("MainWindow", "View", nullptr));
        menuWindow->setTitle(QCoreApplication::translate("MainWindow", "Window", nullptr));
        menuHelp->setTitle(QCoreApplication::translate("MainWindow", "Help", nullptr));
        menuSimulator->setTitle(QCoreApplication::translate("MainWindow", "Simulator", nullptr));
        toolBar->setWindowTitle(QCoreApplication::translate("MainWindow", "toolBar", nullptr));
        dockWidgetPlugins->setWindowTitle(QCoreApplication::translate("MainWindow", "Plugins", nullptr));
        dockWidgetProperties->setWindowTitle(QCoreApplication::translate("MainWindow", "Properties", nullptr));
        dockWidgetConsole->setWindowTitle(QCoreApplication::translate("MainWindow", "Traces", nullptr));
        dockWidgetReports->setWindowTitle(QCoreApplication::translate("MainWindow", "Reports", nullptr));
        dockWidgetExperiments->setWindowTitle(QCoreApplication::translate("MainWindow", "Experiments", nullptr));
    } // retranslateUi

};

namespace Ui {

    class MainWindow : public Ui_MainWindow {
    };
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_MAINWINDOW_H
```

8.19 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_mainwindow.h File Reference

```
#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDockWidget>
#include <QtWidgets/QGraphicsView>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QListWidget>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QMenu>
#include <QtWidgets/QMenuBar>
#include <QtWidgets/QStatusBar>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QTableView>
#include <QtWidgets/QTextEdit>
#include <QtWidgets/QToolBar>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
```

Classes

- class Ui_MainWindow
- class Ui::MainWindow

Namespaces

- Ui

8.20 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/ui_mainwindow.h

```
/********************************************************************************
** Form generated from reading UI file 'mainwindow.ui'
**
** Created by: Qt User Interface Compiler version 5.15.2
**
** WARNING! All changes made in this file will be lost when recompiling UI file!
********************************************************************************/

#ifndef UI_MAINWINDOW_H
#define UI_MAINWINDOW_H

#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDockWidget>
#include <QtWidgets/QGraphicsView>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QListWidget>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QMenu>
#include <QtWidgets/QMenuBar>
#include <QtWidgets/QStatusBar>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QTableView>
#include <QtWidgets/QTextEdit>
#include <QtWidgets/QToolBar>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>

QT_BEGIN_NAMESPACE

class Ui_MainWindow
{
public:
    QAction *actionNew;
    QAction *actionOpen;
    QAction *actionClose;
    QAction *actionSave;
    QAction *actionSave_as;
    QAction *actionExit;
    QAction *actionUndo;
    QAction *actionRedo;
    QAction *actionCut;
    QAction *actionCopy;
    QAction *actionPaste;
    QAction *actionFind;
    QAction *actionReplace;
    QAction *actionGroup;
    QAction *actionZoom_In;
    QAction *actionZoom_Out;
    QAction *actionZoom_Factor;
    QAction *actionGrid;
    QAction *actionSnap;
    QAction *actionInput_Analyser;
    QAction *actionOutput_Analyser;
    QAction *actionScenario_Manager;
    QAction *actionOptimizaton;
    QAction *actionPreferences;
    QAction *actionCheck_Model;
    QAction *actionStart;
    QAction *actionStep;
    QAction *actionStop;
    QAction *actionRun_Control;
    QAction *actionComponents;
    QAction *actionElements;
    QAction *actionModels;
    QAction *actionConsole;
    QAction *actionProperties;
    QAction *actionEditor;
    QAction *actionDocumentation;
    QAction *actionSupport;
    QAction *actionAbout;
    QAction *actionTop;
    QAction *actionBottom;
    QAction *actionInformation;
    QAction *actionReport;
    QAction *actionComponents_2;
    QAction *actionElements_2;
    QAction *actionTrace;
    QAction *actionEvents;
```

```
    QAction *actionPause;
    QAction *actionExit_2;
    QAction *actionPreferences_2;
    QAction *actionModels_2;
    QAction *actionExperiments;
    QAction *actionConnect;
    QAction *actionZoom_In_2;
    QAction *actionZoom_Out_2;
    QAction *actionZoom_Facot;
    QAction *actionGrid_2;
    QAction *actionSnap_2;
    QAction *actionArranje;
    QAction *actionGroup_2;
    QAction *actionPlugins_2;
    QAction *actionInformation_2;
    QAction *actionSimulation;
    QAction *actionComponents_3;
    QAction *actionElements_3;
    QAction *actionLoad;
    QAction *actionSave_2;
    QAction *actionEvents_Diagram;
    QAction *actionArranje_2;
    QWidget *centralWidget;
    QGridLayout *gridLayout;
    QTabWidget *tabWidgetModels;
    QWidget *tabModel1;
    QHBoxLayout *horizontalLayout;
    QGraphicsView *graphicsView;
    QWidget *tabModel2;
    QHBoxLayout *horizontalLayout_2;
    QMenuBar *menuBar;
    QMenu *menuEdit;
    QMenu *menuModel;
    QMenu *menuView_2;
    QMenu *menuWindow;
    QMenu *menuHelp;
    QMenu *menuSimulator;
    QStatusBar *statusBar;
    QToolBar *toolBar;
    QDockWidget *dockWidgetPlugins;
    QWidget *dockWidgetPlugins_2;
    QVBoxLayout *verticalLayout;
    QListWidget *listWidget_Plugins;
    QDockWidget *dockWidgetProperties;
    QWidget *dockWidgetContents_Properties;
    QVBoxLayout *verticalLayout_3;
    QTableView *tableView_Properties;
    QDockWidget *dockWidgetConsole;
    QWidget *dockWidgetContents_Console;
    QVBoxLayout *verticalLayout_4;
    QTextEdit *textEdit_Console;
    QDockWidget *dockWidgetReports;
    QWidget *dockWidgetContents_Reports;
    QVBoxLayout *verticalLayout_5;
    QTextEdit *textEdit_Report;
    QDockWidget *dockWidgetExperiments;
    QWidget *dockWidgetContents_Experiments;
    QListWidget *listWidget_Experiments;

    void setupUi(QMainWindow *MainWindow)
    {
        if (MainWindow->objectName().isEmpty())
            MainWindow->setObjectName(QString::fromUtf8("MainWindow"));
        MainWindow->resize(1273, 724);
        MainWindow->setLocale(QLocale(QLocale::English, QLocale::UnitedStates));
        actionNew = new QAction(MainWindow);
        actionNew->setObjectName(QString::fromUtf8("actionNew"));
        QIcon icon;
        icon.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/60.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionNew->setIcon(icon);
        actionOpen = new QAction(MainWindow);
        actionOpen->setObjectName(QString::fromUtf8("actionOpen"));
        QIcon icon1;
        icon1.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/52.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionOpen->setIcon(icon1);
        actionClose = new QAction(MainWindow);
        actionClose->setObjectName(QString::fromUtf8("actionClose"));
        actionClose->setEnabled(true);
        QIcon icon2;
        icon2.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Close.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionClose->setIcon(icon2);
        actionSave = new QAction(MainWindow);
        actionSave->setObjectName(QString::fromUtf8("actionSave"));
        actionSave->setEnabled(true);
```

```
        QIcon icon3;
        icon3.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Save.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionSave->setIcon(icon3);
        actionSave_as = new QAction(MainWindow);
        actionSave_as->setObjectName(QString::fromUtf8("actionSave_as"));
        actionSave_as->setEnabled(true);
        QIcon icon4;
        icon4.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/86.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionSave_as->setIcon(icon4);
        actionExit = new QAction(MainWindow);
        actionExit->setObjectName(QString::fromUtf8("actionExit"));
        QIcon icon5;
        icon5.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Door.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionExit->setIcon(icon5);
        actionUndo = new QAction(MainWindow);
        actionUndo->setObjectName(QString::fromUtf8("actionUndo"));
        actionUndo->setEnabled(false);
        QIcon icon6;
        icon6.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Undo.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionUndo->setIcon(icon6);
        actionRedo = new QAction(MainWindow);
        actionRedo->setObjectName(QString::fromUtf8("actionRedo"));
        actionRedo->setEnabled(false);
        QIcon icon7;
        icon7.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Redo.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionRedo->setIcon(icon7);
        actionCut = new QAction(MainWindow);
        actionCut->setObjectName(QString::fromUtf8("actionCut"));
        actionCut->setEnabled(false);
        QIcon icon8;
        icon8.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Cut.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionCut->setIcon(icon8);
        actionCopy = new QAction(MainWindow);
        actionCopy->setObjectName(QString::fromUtf8("actionCopy"));
        actionCopy->setEnabled(false);
        QIcon icon9;
        icon9.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Copy.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionCopy->setIcon(icon9);
        actionPaste = new QAction(MainWindow);
        actionPaste->setObjectName(QString::fromUtf8("actionPaste"));
        actionPaste->setEnabled(false);
        QIcon icon10;
        icon10.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Paste.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionPaste->setIcon(icon10);
        actionFind = new QAction(MainWindow);
        actionFind->setObjectName(QString::fromUtf8("actionFind"));
        QIcon icon11;
        icon11.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Find.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionFind->setIcon(icon11);
        actionReplace = new QAction(MainWindow);
        actionReplace->setObjectName(QString::fromUtf8("actionReplace"));
        QIcon icon12;
        icon12.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/View.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionReplace->setIcon(icon12);
        actionGroup = new QAction(MainWindow);
        actionGroup->setObjectName(QString::fromUtf8("actionGroup"));
        actionZoom_In = new QAction(MainWindow);
        actionZoom_In->setObjectName(QString::fromUtf8("actionZoom_In"));
        QIcon icon13;
        icon13.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Zoom.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionZoom_In->setIcon(icon13);
        actionZoom_Out = new QAction(MainWindow);
        actionZoom_Out->setObjectName(QString::fromUtf8("actionZoom_Out"));
        actionZoom_Out->setIcon(icon13);
        actionZoom_Factor = new QAction(MainWindow);
        actionZoom_Factor->setObjectName(QString::fromUtf8("actionZoom_Factor"));
        actionZoom_Factor->setIcon(icon13);
        actionGrid = new QAction(MainWindow);
        actionGrid->setObjectName(QString::fromUtf8("actionGrid"));
        actionSnap = new QAction(MainWindow);
        actionSnap->setObjectName(QString::fromUtf8("actionSnap"));
        actionInput_Analyser = new QAction(MainWindow);
        actionInput_Analyser->setObjectName(QString::fromUtf8("actionInput_Analyser"));
        actionOutput_Analyser = new QAction(MainWindow);
        actionOutput_Analyser->setObjectName(QString::fromUtf8("actionOutput_Analyser"));
        actionScenario_Manager = new QAction(MainWindow);
```

```
        actionScenario_Manager->setObjectName(QString::fromUtf8("actionScenario_Manager"));
        actionOptimizaton = new QAction(MainWindow);
        actionOptimizaton->setObjectName(QString::fromUtf8("actionOptimizaton"));
        actionPreferences = new QAction(MainWindow);
        actionPreferences->setObjectName(QString::fromUtf8("actionPreferences"));
        actionCheck_Model = new QAction(MainWindow);
        actionCheck_Model->setObjectName(QString::fromUtf8("actionCheck_Model"));
        QIcon icon14;
        icon14.addFile(QString::fromUtf8(":/rec/resources/icons/pack2/png/24x24/58.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionCheck_Model->setIcon(icon14);
        actionStart = new QAction(MainWindow);
        actionStart->setObjectName(QString::fromUtf8("actionStart"));
        QIcon icon15;
        icon15.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Play.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionStart->setIcon(icon15);
        actionStep = new QAction(MainWindow);
        actionStep->setObjectName(QString::fromUtf8("actionStep"));
        QIcon icon16;
        icon16.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Next track.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionStep->setIcon(icon16);
        actionStop = new QAction(MainWindow);
        actionStop->setObjectName(QString::fromUtf8("actionStop"));
        QIcon icon17;
        icon17.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Stop.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionStop->setIcon(icon17);
        actionRun_Control = new QAction(MainWindow);
        actionRun_Control->setObjectName(QString::fromUtf8("actionRun_Control"));
        QIcon icon18;
        icon18.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Tune.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionRun_Control->setIcon(icon18);
        actionComponents = new QAction(MainWindow);
        actionComponents->setObjectName(QString::fromUtf8("actionComponents"));
        actionElements = new QAction(MainWindow);
        actionElements->setObjectName(QString::fromUtf8("actionElements"));
        actionModels = new QAction(MainWindow);
        actionModels->setObjectName(QString::fromUtf8("actionModels"));
        actionConsole = new QAction(MainWindow);
        actionConsole->setObjectName(QString::fromUtf8("actionConsole"));
        actionProperties = new QAction(MainWindow);
        actionProperties->setObjectName(QString::fromUtf8("actionProperties"));
        actionEditor = new QAction(MainWindow);
        actionEditor->setObjectName(QString::fromUtf8("actionEditor"));
        actionDocumentation = new QAction(MainWindow);
        actionDocumentation->setObjectName(QString::fromUtf8("actionDocumentation"));
        QIcon icon19;
        icon19.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Help book.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionDocumentation->setIcon(icon19);
        actionSupport = new QAction(MainWindow);
        actionSupport->setObjectName(QString::fromUtf8("actionSupport"));
        QIcon icon20;
        icon20.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Help.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionSupport->setIcon(icon20);
        actionAbout = new QAction(MainWindow);
        actionAbout->setObjectName(QString::fromUtf8("actionAbout"));
        QIcon icon21;
        icon21.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Help book 3d.png"), QSize(), QIcon::Normal, QIcon::Off);
        actionAbout->setIcon(icon21);
        actionTop = new QAction(MainWindow);
        actionTop->setObjectName(QString::fromUtf8("actionTop"));
        actionBottom = new QAction(MainWindow);
        actionBottom->setObjectName(QString::fromUtf8("actionBottom"));
        actionInformation = new QAction(MainWindow);
        actionInformation->setObjectName(QString::fromUtf8("actionInformation"));
        actionInformation->setEnabled(true);
        actionReport = new QAction(MainWindow);
        actionReport->setObjectName(QString::fromUtf8("actionReport"));
        actionComponents_2 = new QAction(MainWindow);
        actionComponents_2->setObjectName(QString::fromUtf8("actionComponents_2"));
        actionElements_2 = new QAction(MainWindow);
        actionElements_2->setObjectName(QString::fromUtf8("actionElements_2"));
        actionTrace = new QAction(MainWindow);
        actionTrace->setObjectName(QString::fromUtf8("actionTrace"));
        actionEvents = new QAction(MainWindow);
        actionEvents->setObjectName(QString::fromUtf8("actionEvents"));
        actionPause = new QAction(MainWindow);
        actionPause->setObjectName(QString::fromUtf8("actionPause"));
        QIcon icon22;
        icon22.addFile(QString::fromUtf8(":/rec/resources/icons/pack3/png/24x24/Pause.png"), QSize(), QIcon::Normal, QIcon::Off);
```

```
        actionPause->setIcon(icon22);
        actionExit_2 = new QAction(MainWindow);
        actionExit_2->setObjectName(QString::fromUtf8("actionExit_2"));
        actionPreferences_2 = new QAction(MainWindow);
        actionPreferences_2->setObjectName(QString::fromUtf8("actionPreferences_2"));
        actionModels_2 = new QAction(MainWindow);
        actionModels_2->setObjectName(QString::fromUtf8("actionModels_2"));
        actionExperiments = new QAction(MainWindow);
        actionExperiments->setObjectName(QString::fromUtf8("actionExperiments"));
        actionConnect = new QAction(MainWindow);
        actionConnect->setObjectName(QString::fromUtf8("actionConnect"));
        actionZoom_In_2 = new QAction(MainWindow);
        actionZoom_In_2->setObjectName(QString::fromUtf8("actionZoom_In_2"));
        actionZoom_Out_2 = new QAction(MainWindow);
        actionZoom_Out_2->setObjectName(QString::fromUtf8("actionZoom_Out_2"));
        actionZoom_Facot = new QAction(MainWindow);
        actionZoom_Facot->setObjectName(QString::fromUtf8("actionZoom_Facot"));
        actionGrid_2 = new QAction(MainWindow);
        actionGrid_2->setObjectName(QString::fromUtf8("actionGrid_2"));
        actionSnap_2 = new QAction(MainWindow);
        actionSnap_2->setObjectName(QString::fromUtf8("actionSnap_2"));
        actionArranje = new QAction(MainWindow);
        actionArranje->setObjectName(QString::fromUtf8("actionArranje"));
        actionGroup_2 = new QAction(MainWindow);
        actionGroup_2->setObjectName(QString::fromUtf8("actionGroup_2"));
        actionPlugins_2 = new QAction(MainWindow);
        actionPlugins_2->setObjectName(QString::fromUtf8("actionPlugins_2"));
        actionInformation_2 = new QAction(MainWindow);
        actionInformation_2->setObjectName(QString::fromUtf8("actionInformation_2"));
        actionSimulation = new QAction(MainWindow);
        actionSimulation->setObjectName(QString::fromUtf8("actionSimulation"));
        actionComponents_3 = new QAction(MainWindow);
        actionComponents_3->setObjectName(QString::fromUtf8("actionComponents_3"));
        actionElements_3 = new QAction(MainWindow);
        actionElements_3->setObjectName(QString::fromUtf8("actionElements_3"));
        actionLoad = new QAction(MainWindow);
        actionLoad->setObjectName(QString::fromUtf8("actionLoad"));
        actionSave_2 = new QAction(MainWindow);
        actionSave_2->setObjectName(QString::fromUtf8("actionSave_2"));
        actionEvents_Diagram = new QAction(MainWindow);
        actionEvents_Diagram->setObjectName(QString::fromUtf8("actionEvents_Diagram"));
        actionArranje_2 = new QAction(MainWindow);
        actionArranje_2->setObjectName(QString::fromUtf8("actionArranje_2"));
        centralWidget = new QWidget(MainWindow);
        centralWidget->setObjectName(QString::fromUtf8("centralWidget"));
        gridLayout = new QGridLayout(centralWidget);
        gridLayout->setSpacing(6);
        gridLayout->setContentsMargins(11, 11, 11, 11);
        gridLayout->setObjectName(QString::fromUtf8("gridLayout"));
        tabWidgetModels = new QTabWidget(centralWidget);
        tabWidgetModels->setObjectName(QString::fromUtf8("tabWidgetModels"));
        tabWidgetModels->setTabPosition(QTabWidget::North);
        tabWidgetModels->setTabShape(QTabWidget::Rounded);
        tabWidgetModels->setTabsClosable(true);
        tabWidgetModels->setMovable(true);
        tabModel1 = new QWidget();
        tabModel1->setObjectName(QString::fromUtf8("tabModel1"));
        horizontalLayout = new QHBoxLayout(tabModel1);
        horizontalLayout->setSpacing(6);
        horizontalLayout->setContentsMargins(11, 11, 11, 11);
        horizontalLayout->setObjectName(QString::fromUtf8("horizontalLayout"));
        graphicsView = new QGraphicsView(tabModel1);
        graphicsView->setObjectName(QString::fromUtf8("graphicsView"));
        graphicsView->setFrameShape(QFrame::WinPanel);
        graphicsView->setFrameShadow(QFrame::Sunken);
        graphicsView->setLineWidth(2);
        graphicsView->setMidLineWidth(1);
        graphicsView->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        graphicsView->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        QBrush brush(QColor(238, 238, 238, 255));
        brush.setStyle(Qt::SolidPattern);
        graphicsView->setBackgroundBrush(brush);
        graphicsView->setDragMode(QGraphicsView::NoDrag);

        horizontalLayout->addWidget(graphicsView);

        tabWidgetModels->addTab(tabModel1, QString());
        tabModel2 = new QWidget();
        tabModel2->setObjectName(QString::fromUtf8("tabModel2"));
        horizontalLayout_2 = new QHBoxLayout(tabModel2);
        horizontalLayout_2->setSpacing(6);
        horizontalLayout_2->setContentsMargins(11, 11, 11, 11);
        horizontalLayout_2->setObjectName(QString::fromUtf8("horizontalLayout_2"));
        tabWidgetModels->addTab(tabModel2, QString());

        gridLayout->addWidget(tabWidgetModels, 0, 0, 1, 1);

```

```
        MainWindow->setCentralWidget(centralWidget);
        menuBar = new QMenuBar(MainWindow);
        menuBar->setObjectName(QString::fromUtf8("menuBar"));
        menuBar->setGeometry(QRect(0, 0, 1273, 24));
        menuEdit = new QMenu(menuBar);
        menuEdit->setObjectName(QString::fromUtf8("menuEdit"));
        menuModel = new QMenu(menuBar);
        menuModel->setObjectName(QString::fromUtf8("menuModel"));
        menuModel->setEnabled(true);
        menuView_2 = new QMenu(menuModel);
        menuView_2->setObjectName(QString::fromUtf8("menuView_2"));
        menuWindow = new QMenu(menuBar);
        menuWindow->setObjectName(QString::fromUtf8("menuWindow"));
        menuHelp = new QMenu(menuBar);
        menuHelp->setObjectName(QString::fromUtf8("menuHelp"));
        menuSimulator = new QMenu(menuBar);
        menuSimulator->setObjectName(QString::fromUtf8("menuSimulator"));
        MainWindow->setMenuBar(menuBar);
        statusBar = new QStatusBar(MainWindow);
        statusBar->setObjectName(QString::fromUtf8("statusBar"));
        MainWindow->setStatusBar(statusBar);
        toolBar = new QToolBar(MainWindow);
        toolBar->setObjectName(QString::fromUtf8("toolBar"));
        MainWindow->addToolBar(Qt::TopToolBarArea, toolBar);
        dockWidgetPlugins = new QDockWidget(MainWindow);
        dockWidgetPlugins->setObjectName(QString::fromUtf8("dockWidgetPlugins"));
        dockWidgetPlugins->setAutoFillBackground(false);

dockWidgetPlugins->setFeatures(QDockWidget::DockWidgetFloatable|QDockWidget::DockWidgetMovable);
        dockWidgetPlugins_2 = new QWidget();
        dockWidgetPlugins_2->setObjectName(QString::fromUtf8("dockWidgetPlugins_2"));
        verticalLayout = new QVBoxLayout(dockWidgetPlugins_2);
        verticalLayout->setSpacing(6);
        verticalLayout->setContentsMargins(11, 11, 11, 11);
        verticalLayout->setObjectName(QString::fromUtf8("verticalLayout"));
        listWidget_Plugins = new QListWidget(dockWidgetPlugins_2);
        listWidget_Plugins->setObjectName(QString::fromUtf8("listWidget_Plugins"));

        verticalLayout->addWidget(listWidget_Plugins);

        dockWidgetPlugins->setWidget(dockWidgetPlugins_2);
        MainWindow->addDockWidget(Qt::LeftDockWidgetArea, dockWidgetPlugins);
        dockWidgetProperties = new QDockWidget(MainWindow);
        dockWidgetProperties->setObjectName(QString::fromUtf8("dockWidgetProperties"));

dockWidgetProperties->setFeatures(QDockWidget::DockWidgetFloatable|QDockWidget::DockWidgetMovable);
        dockWidgetContents_Properties = new QWidget();

dockWidgetContents_Properties->setObjectName(QString::fromUtf8("dockWidgetContents_Properties"));
        verticalLayout_3 = new QVBoxLayout(dockWidgetContents_Properties);
        verticalLayout_3->setSpacing(6);
        verticalLayout_3->setContentsMargins(11, 11, 11, 11);
        verticalLayout_3->setObjectName(QString::fromUtf8("verticalLayout_3"));
        tableView_Properties = new QTableView(dockWidgetContents_Properties);
        tableView_Properties->setObjectName(QString::fromUtf8("tableView_Properties"));
        tableView_Properties->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        tableView_Properties->setHorizontalScrollBarPolicy(Qt::ScrollBarAsNeeded);

        verticalLayout_3->addWidget(tableView_Properties);

        dockWidgetProperties->setWidget(dockWidgetContents_Properties);
        MainWindow->addDockWidget(Qt::RightDockWidgetArea, dockWidgetProperties);
        dockWidgetConsole = new QDockWidget(MainWindow);
        dockWidgetConsole->setObjectName(QString::fromUtf8("dockWidgetConsole"));
        dockWidgetConsole->setFloating(false);

dockWidgetConsole->setFeatures(QDockWidget::DockWidgetFloatable|QDockWidget::DockWidgetMovable);
        dockWidgetConsole->setAllowedAreas(Qt::AllDockWidgetAreas);
        dockWidgetContents_Console = new QWidget();
        dockWidgetContents_Console->setObjectName(QString::fromUtf8("dockWidgetContents_Console"));
        verticalLayout_4 = new QVBoxLayout(dockWidgetContents_Console);
        verticalLayout_4->setSpacing(6);
        verticalLayout_4->setContentsMargins(11, 11, 11, 11);
        verticalLayout_4->setObjectName(QString::fromUtf8("verticalLayout_4"));
        textEdit_Console = new QTextEdit(dockWidgetContents_Console);
        textEdit_Console->setObjectName(QString::fromUtf8("textEdit_Console"));
        QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
        sizePolicy.setHorizontalStretch(0);
        sizePolicy.setVerticalStretch(0);
        sizePolicy.setHeightForWidth(textEdit_Console->sizePolicy().hasHeightForWidth());
        textEdit_Console->setSizePolicy(sizePolicy);
        textEdit_Console->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
        textEdit_Console->setLineWrapMode(QTextEdit::NoWrap);
        textEdit_Console->setReadOnly(true);
        textEdit_Console->setTextInteractionFlags(Qt::NoTextInteraction);

        verticalLayout_4->addWidget(textEdit_Console);
```

```
        dockWidgetConsole->setWidget(dockWidgetContents_Console);
        MainWindow->addDockWidget(Qt::BottomDockWidgetArea, dockWidgetConsole);
        dockWidgetReports = new QDockWidget(MainWindow);
        dockWidgetReports->setObjectName(QString::fromUtf8("dockWidgetReports"));
dockWidgetReports->setFeatures(QDockWidget::DockWidgetFloatable|QDockWidget::DockWidgetMovable);
        dockWidgetContents_Reports = new QWidget();
        dockWidgetContents_Reports->setObjectName(QString::fromUtf8("dockWidgetContents_Reports"));
        verticalLayout_5 = new QVBoxLayout(dockWidgetContents_Reports);
        verticalLayout_5->setSpacing(6);
        verticalLayout_5->setContentsMargins(11, 11, 11, 11);
        verticalLayout_5->setObjectName(QString::fromUtf8("verticalLayout_5"));
        textEdit_Report = new QTextEdit(dockWidgetContents_Reports);
        textEdit_Report->setObjectName(QString::fromUtf8("textEdit_Report"));
        sizePolicy.setHeightForWidth(textEdit_Report->sizePolicy().hasHeightForWidth());
        textEdit_Report->setSizePolicy(sizePolicy);
        textEdit_Report->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOn);

        verticalLayout_5->addWidget(textEdit_Report);

        dockWidgetReports->setWidget(dockWidgetContents_Reports);
        MainWindow->addDockWidget(Qt::BottomDockWidgetArea, dockWidgetReports);
        dockWidgetExperiments = new QDockWidget(MainWindow);
        dockWidgetExperiments->setObjectName(QString::fromUtf8("dockWidgetExperiments"));
        dockWidgetContents_Experiments = new QWidget();
dockWidgetContents_Experiments->setObjectName(QString::fromUtf8("dockWidgetContents_Experiments"));
        dockWidgetContents_Experiments->setMinimumSize(QSize(0, 144));
        listWidget_Experiments = new QListWidget(dockWidgetContents_Experiments);
        listWidget_Experiments->setObjectName(QString::fromUtf8("listWidget_Experiments"));
        listWidget_Experiments->setGeometry(QRect(10, 10, 201, 161));
        listWidget_Experiments->setAutoFillBackground(false);
        dockWidgetExperiments->setWidget(dockWidgetContents_Experiments);
        MainWindow->addDockWidget(Qt::LeftDockWidgetArea, dockWidgetExperiments);

        menuBar->addAction(menuSimulator->menuAction());
        menuBar->addAction(menuEdit->menuAction());
        menuBar->addAction(menuModel->menuAction());
        menuBar->addAction(menuWindow->menuAction());
        menuBar->addAction(menuHelp->menuAction());
        menuEdit->addAction(actionUndo);
        menuEdit->addAction(actionRedo);
        menuEdit->addSeparator();
        menuEdit->addAction(actionCut);
        menuEdit->addAction(actionCopy);
        menuEdit->addAction(actionPaste);
        menuEdit->addSeparator();
        menuEdit->addAction(actionFind);
        menuEdit->addAction(actionReplace);
        menuModel->addAction(actionInformation_2);
        menuModel->addAction(actionSimulation);
        menuModel->addSeparator();
        menuModel->addAction(actionComponents_3);
        menuModel->addAction(actionElements_3);
        menuModel->addSeparator();
        menuModel->addAction(actionSave_2);
        menuModel->addSeparator();
        menuModel->addAction(menuView_2->menuAction());
        menuView_2->addAction(actionZoom_In_2);
        menuView_2->addAction(actionZoom_Out_2);
        menuView_2->addAction(actionZoom_Facot);
        menuView_2->addSeparator();
        menuView_2->addAction(actionGrid_2);
        menuView_2->addAction(actionSnap_2);
        menuView_2->addSeparator();
        menuView_2->addAction(actionArranje);
        menuView_2->addAction(actionGroup_2);
        menuWindow->addAction(actionComponents);
        menuWindow->addAction(actionElements);
        menuWindow->addAction(actionProperties);
        menuWindow->addAction(actionEditor);
        menuWindow->addAction(actionConsole);
        menuWindow->addSeparator();
        menuWindow->addAction(actionModels);
        menuWindow->addSeparator();
        menuWindow->addAction(actionArranje_2);
        menuHelp->addAction(actionDocumentation);
        menuHelp->addAction(actionSupport);
        menuHelp->addSeparator();
        menuHelp->addAction(actionAbout);
        menuSimulator->addSeparator();
        menuSimulator->addAction(actionPlugins_2);
        menuSimulator->addAction(actionExperiments);
        menuSimulator->addAction(actionModels_2);
        menuSimulator->addSeparator();
        menuSimulator->addAction(actionPreferences_2);
```

```
        menuSimulator->addSeparator();
        menuSimulator->addAction(actionExit_2);
        toolBar->addAction(actionZoom_Out);
        toolBar->addAction(actionZoom_Factor);

        retranslateUi(MainWindow);

        tabWidgetModels->setCurrentIndex(0);


        QMetaObject::connectSlotsByName(MainWindow);
    } // setupUi

    void retranslateUi(QMainWindow *MainWindow)
    {
        MainWindow->setWindowTitle(QCoreApplication::translate("MainWindow", "GenESyS -- Generic and Expansible System Simulator", nullptr));
        actionNew->setText(QCoreApplication::translate("MainWindow", "New", nullptr));
#if QT_CONFIG(tooltip)
        actionNew->setToolTip(QCoreApplication::translate("MainWindow", "New simulation model", nullptr));
#endif // QT_CONFIG(tooltip)
        actionOpen->setText(QCoreApplication::translate("MainWindow", "Open", nullptr));
#if QT_CONFIG(shortcut)
        actionOpen->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+O", nullptr));
#endif // QT_CONFIG(shortcut)
        actionClose->setText(QCoreApplication::translate("MainWindow", "Close", nullptr));
#if QT_CONFIG(shortcut)
        actionClose->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+W", nullptr));
#endif // QT_CONFIG(shortcut)
        actionSave->setText(QCoreApplication::translate("MainWindow", "Save", nullptr));
#if QT_CONFIG(shortcut)
        actionSave->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+S", nullptr));
#endif // QT_CONFIG(shortcut)
        actionSave_as->setText(QCoreApplication::translate("MainWindow", "Save all", nullptr));
#if QT_CONFIG(shortcut)
        actionSave_as->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+A", nullptr));
#endif // QT_CONFIG(shortcut)
        actionExit->setText(QCoreApplication::translate("MainWindow", "Exit", nullptr));
#if QT_CONFIG(shortcut)
        actionExit->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+F4", nullptr));
#endif // QT_CONFIG(shortcut)
        actionUndo->setText(QCoreApplication::translate("MainWindow", "Undo", nullptr));
#if QT_CONFIG(shortcut)
        actionUndo->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+Z", nullptr));
#endif // QT_CONFIG(shortcut)
        actionRedo->setText(QCoreApplication::translate("MainWindow", "Redo", nullptr));
#if QT_CONFIG(shortcut)
        actionRedo->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+Shift+Z", nullptr));
#endif // QT_CONFIG(shortcut)
        actionCut->setText(QCoreApplication::translate("MainWindow", "Cut", nullptr));
#if QT_CONFIG(shortcut)
        actionCut->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+X", nullptr));
#endif // QT_CONFIG(shortcut)
        actionCopy->setText(QCoreApplication::translate("MainWindow", "Copy", nullptr));
#if QT_CONFIG(shortcut)
        actionCopy->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+C", nullptr));
#endif // QT_CONFIG(shortcut)
        actionPaste->setText(QCoreApplication::translate("MainWindow", "Paste", nullptr));
#if QT_CONFIG(shortcut)
        actionPaste->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+V", nullptr));
#endif // QT_CONFIG(shortcut)
        actionFind->setText(QCoreApplication::translate("MainWindow", "Find", nullptr));
#if QT_CONFIG(shortcut)
        actionFind->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+F", nullptr));
#endif // QT_CONFIG(shortcut)
        actionReplace->setText(QCoreApplication::translate("MainWindow", "Replace", nullptr));
#if QT_CONFIG(shortcut)
        actionReplace->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+H", nullptr));
#endif // QT_CONFIG(shortcut)
        actionGroup->setText(QCoreApplication::translate("MainWindow", "Group", nullptr));
        actionZoom_In->setText(QCoreApplication::translate("MainWindow", "Zoom In", nullptr));
#if QT_CONFIG(shortcut)
        actionZoom_In->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+=", nullptr));
#endif // QT_CONFIG(shortcut)
        actionZoom_Out->setText(QCoreApplication::translate("MainWindow", "Zoom Out", nullptr));
#if QT_CONFIG(shortcut)
        actionZoom_Out->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+-", nullptr));
#endif // QT_CONFIG(shortcut)
        actionZoom_Factor->setText(QCoreApplication::translate("MainWindow", "Zoom Factor", nullptr));
        actionGrid->setText(QCoreApplication::translate("MainWindow", "Grid", nullptr));
        actionSnap->setText(QCoreApplication::translate("MainWindow", "Snap", nullptr));
        actionInput_Analyser->setText(QCoreApplication::translate("MainWindow", "Input Analyser", nullptr));
        actionOutput_Analyser->setText(QCoreApplication::translate("MainWindow", "Output Analyser", nullptr));
        actionScenario_Manager->setText(QCoreApplication::translate("MainWindow", "Scenario Manager",
```

```
nullptr));
        actionOptimizaton->setText(QCoreApplication::translate("MainWindow", "Optimizaton", nullptr));
        actionPreferences->setText(QCoreApplication::translate("MainWindow", "Preferences", nullptr));
        actionCheck_Model->setText(QCoreApplication::translate("MainWindow", "Check", nullptr));
#if QT_CONFIG(shortcut)
        actionCheck_Model->setShortcut(QCoreApplication::translate("MainWindow", "F4", nullptr));
#endif // QT_CONFIG(shortcut)
        actionStart->setText(QCoreApplication::translate("MainWindow", "Start", nullptr));
#if QT_CONFIG(shortcut)
        actionStart->setShortcut(QCoreApplication::translate("MainWindow", "F5", nullptr));
#endif // QT_CONFIG(shortcut)
        actionStep->setText(QCoreApplication::translate("MainWindow", "Step", nullptr));
#if QT_CONFIG(shortcut)
        actionStep->setShortcut(QCoreApplication::translate("MainWindow", "F7", nullptr));
#endif // QT_CONFIG(shortcut)
        actionStop->setText(QCoreApplication::translate("MainWindow", "Stop", nullptr));
#if QT_CONFIG(shortcut)
        actionStop->setShortcut(QCoreApplication::translate("MainWindow", "F10", nullptr));
#endif // QT_CONFIG(shortcut)
        actionRun_Control->setText(QCoreApplication::translate("MainWindow", "Settings", nullptr));
        actionComponents->setText(QCoreApplication::translate("MainWindow", "Plugins", nullptr));
        actionElements->setText(QCoreApplication::translate("MainWindow", "Experiments", nullptr));
        actionModels->setText(QCoreApplication::translate("MainWindow", "Models", nullptr));
        actionConsole->setText(QCoreApplication::translate("MainWindow", "Console", nullptr));
        actionProperties->setText(QCoreApplication::translate("MainWindow", "Properties", nullptr));
        actionEditor->setText(QCoreApplication::translate("MainWindow", "Reports", nullptr));
        actionDocumentation->setText(QCoreApplication::translate("MainWindow", "Documentation",
nullptr));
#if QT_CONFIG(shortcut)
        actionDocumentation->setShortcut(QCoreApplication::translate("MainWindow", "F1", nullptr));
#endif // QT_CONFIG(shortcut)
        actionSupport->setText(QCoreApplication::translate("MainWindow", "Support", nullptr));
#if QT_CONFIG(shortcut)
        actionSupport->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+F1", nullptr));
#endif // QT_CONFIG(shortcut)
        actionAbout->setText(QCoreApplication::translate("MainWindow", "About...", nullptr));
#if QT_CONFIG(shortcut)
        actionAbout->setShortcut(QCoreApplication::translate("MainWindow", "Ctrl+Shift+F1", nullptr));
#endif // QT_CONFIG(shortcut)
        actionTop->setText(QCoreApplication::translate("MainWindow", "Top", nullptr));
        actionBottom->setText(QCoreApplication::translate("MainWindow", "Bottom", nullptr));
        actionInformation->setText(QCoreApplication::translate("MainWindow", "Information", nullptr));
        actionReport->setText(QCoreApplication::translate("MainWindow", "Report", nullptr));
        actionComponents_2->setText(QCoreApplication::translate("MainWindow", "Components", nullptr));
        actionElements_2->setText(QCoreApplication::translate("MainWindow", "Elements", nullptr));
        actionTrace->setText(QCoreApplication::translate("MainWindow", "Trace", nullptr));
        actionEvents->setText(QCoreApplication::translate("MainWindow", "Events", nullptr));
        actionPause->setText(QCoreApplication::translate("MainWindow", "Pause", nullptr));
        actionExit_2->setText(QCoreApplication::translate("MainWindow", "Exit", nullptr));
        actionPreferences_2->setText(QCoreApplication::translate("MainWindow", "Preferences",
nullptr));
        actionModels_2->setText(QCoreApplication::translate("MainWindow", "Models", nullptr));
        actionExperiments->setText(QCoreApplication::translate("MainWindow", "Experiments", nullptr));
        actionConnect->setText(QCoreApplication::translate("MainWindow", "Connect", nullptr));
        actionZoom_In_2->setText(QCoreApplication::translate("MainWindow", "Zoom In", nullptr));
        actionZoom_Out_2->setText(QCoreApplication::translate("MainWindow", "Zoom Out", nullptr));
        actionZoom_Facot->setText(QCoreApplication::translate("MainWindow", "Zoom Facot", nullptr));
        actionGrid_2->setText(QCoreApplication::translate("MainWindow", "Grid", nullptr));
        actionSnap_2->setText(QCoreApplication::translate("MainWindow", "Snap", nullptr));
        actionArranje->setText(QCoreApplication::translate("MainWindow", "Arranje", nullptr));
        actionGroup_2->setText(QCoreApplication::translate("MainWindow", "Group", nullptr));
        actionPlugins_2->setText(QCoreApplication::translate("MainWindow", "Plugins", nullptr));
        actionInformation_2->setText(QCoreApplication::translate("MainWindow", "Information",
nullptr));
        actionSimulation->setText(QCoreApplication::translate("MainWindow", "Simulation", nullptr));
        actionComponents_3->setText(QCoreApplication::translate("MainWindow", "Components", nullptr));
        actionElements_3->setText(QCoreApplication::translate("MainWindow", "Elements", nullptr));
        actionLoad->setText(QCoreApplication::translate("MainWindow", "Load", nullptr));
        actionSave_2->setText(QCoreApplication::translate("MainWindow", "Save", nullptr));
        actionEvents_Diagram->setText(QCoreApplication::translate("MainWindow", "Events Diagram",
nullptr));
        actionArranje_2->setText(QCoreApplication::translate("MainWindow", "Arranje", nullptr));
        tabWidgetModels->setTabText(tabWidgetModels->indexOf(tabModel1),
QCoreApplication::translate("MainWindow", "Model 1", nullptr));
        tabWidgetModels->setTabText(tabWidgetModels->indexOf(tabModel2),
QCoreApplication::translate("MainWindow", "Model 2", nullptr));
        menuEdit->setTitle(QCoreApplication::translate("MainWindow", "Edit", nullptr));
        menuModel->setTitle(QCoreApplication::translate("MainWindow", "Model", nullptr));
        menuView_2->setTitle(QCoreApplication::translate("MainWindow", "View", nullptr));
        menuWindow->setTitle(QCoreApplication::translate("MainWindow", "Window", nullptr));
        menuHelp->setTitle(QCoreApplication::translate("MainWindow", "Help", nullptr));
        menuSimulator->setTitle(QCoreApplication::translate("MainWindow", "Simulator", nullptr));
        toolBar->setWindowTitle(QCoreApplication::translate("MainWindow", "toolBar", nullptr));
        dockWidgetPlugins->setWindowTitle(QCoreApplication::translate("MainWindow", "Plugins",
nullptr));
        dockWidgetProperties->setWindowTitle(QCoreApplication::translate("MainWindow", "Properties",
nullptr));
```

```
        dockWidgetConsole->setWindowTitle(QCoreApplication::translate("MainWindow", "Traces", nullptr));
        dockWidgetReports->setWindowTitle(QCoreApplication::translate("MainWindow", "Reports", nullptr));
        dockWidgetExperiments->setWindowTitle(QCoreApplication::translate("MainWindow", "Experiments", nullptr));
    } // retranslateUi

};

namespace Ui {
    class MainWindow: public Ui_MainWindow {};
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_MAINWINDOW_H
```

8.21 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mapkernelandui.cpp File Reference

`#include "mapkernelandui.h"`

Include dependency graph for mapkernelandui.cpp:

8.22 mapkernelandui.cpp

```

#include "mapkernelandui.h"

MapKernelAndUI::MapKernelAndUI() {

}

void MapKernelAndUI::map(Model* model, QObject* uiObj) {
    _mapModelQObj.insert({model, uiObj});
    _mapQObjModel.insert({uiObj, model});
}

void MapKernelAndUI::map(ModelElement* elem, QObject* uiObj) {
    _mapElemQObj.insert({elem, uiObj});
    _mapQObjElem.insert({uiObj, elem});
}

void MapKernelAndUI::unmap(Model* model, QObject* uiObj) {
    std::map<Model*, QObject*>::iterator it = _mapModelQObj.find(model);
    if (it != _mapModelQObj.end())
        _mapModelQObj.erase(it);
```

```
    std::map<QObject*, Model*>::iterator it2 = _mapQObjModel.find(uiObj);
    if (it2 != _mapQObjModel.end())
        _mapQObjModel.erase(it2);
}

void MapKernelAndUI::unmap(ModelElement* elem, QObject* uiObj) {
    std::map<ModelElement*, QObject*>::iterator it = _mapElemQObj.find(elem);
    if (it != _mapElemQObj.end())
        _mapElemQObj.erase(it);
    std::map<QObject*, ModelElement*>::iterator it2 = _mapQObjElem.find(uiObj);
    if (it2 != _mapQObjElem.end())
        _mapQObjElem.erase(it2);
}

QObject* MapKernelAndUI::getUI(Model* model) {
    std::map<Model*, QObject*>::iterator it = _mapModelQObj.find(model);
    if (it != _mapModelQObj.end())
        return (*it).second;
    return nullptr;
}

QObject* MapKernelAndUI::getUI(ModelElement* elem) {
    std::map<ModelElement*, QObject*>::iterator it = _mapElemQObj.find(elem);
    if (it != _mapElemQObj.end())
        return (*it).second;
    return nullptr;
}

Model* MapKernelAndUI::getModel(QObject* uiObj) {
    std::map<QObject*, Model*>::iterator it = _mapQObjModel.find(uiObj);
    if (it != _mapQObjModel.end())
        return (*it).second;
    return nullptr;
}

ModelElement* MapKernelAndUI::getElem(QObject* uiObj) {
    std::map<QObject*, ModelElement*>::iterator it = _mapQObjElem.find(uiObj);
    if (it != _mapQObjElem.end())
        return (*it).second;
    return nullptr;
}
```

8.23 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/gui/qt/mapkernelandui.h File Reference

```
#include <map>
#include "../../../kernel/simulator/Model.h"
#include "../../../kernel/simulator/ModelElement.h"
#include <QObject>
```

Include dependency graph for mapkernelandui.h:

This graph shows which files directly or indirectly include this file:

Classes

- class MapKernelAndUI

8.24 mapkernelandui.h

```
#ifndef MAPKERNELANDUI_H
#define MAPKERNELANDUI_H

#include <map>
#include "../../../kernel/simulator/Model.h"
#include "../../../kernel/simulator/ModelElement.h"
#include <QObject>

class MapKernelAndUI {
public:
    MapKernelAndUI();
public:
    void map(Model* model, QObject* uiObj);
    void map(ModelElement* elem, QObject* uiObj);
    void unmap(Model* model, QObject* uiObj);
    void unmap(ModelElement* elem, QObject* uiObj);
    QObject* getUI(Model* model);
    QObject* getUI(ModelElement* elem);
    Model* getModel(QObject* uiObj);
    ModelElement* getElem(QObject* uiObj);
private:
    std::map<Model*, QObject*> _mapModelQObj;
    std::map<ModelElement*, QObject*> _mapElemQObj;
    std::map<QObject*, Model*> _mapQObjModel;
    std::map<QObject*, ModelElement*> _mapQObjElem;
};

#endif // MAPKERNELANDUI_H
```

8.25 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/terminal/GenesysShell_if.h File Reference

```
#include "GenesysApplication_if.h"
#include "Util.h"
```

Include dependency graph for GenesysShell_if.h:

Classes

- class GenesysShell_if

8.26 GenesysShell_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   GenesysShell_if.h
 * Author: rlcancian
 *
 * Created on 24 de Maio de 2019, 11:02
 */

#ifndef GENESYSSHELL_IF_H
#define GENESYSSHELL_IF_H

#include "GenesysApplication_if.h"
#include "Util.h"

class GenesysShell_if : public GenesysApplication_if {
public:
    virtual void openModel(std::string filename) = 0;
    virtual void saveModelAs(std::string filename) = 0;
    virtual void saveModel() = 0;
    virtual void listElements() = 0;
    virtual void listComponents() = 0;
    virtual void listHosts() = 0;
    virtual void listPlugins() = 0;
    virtual void deleteTraceFiles() = 0;
    virtual void traceLevel(Util::TraceLevel tracelevel) = 0;
    virtual void addPlugin(std::string filename) = 0;
    virtual void addFromFile(std::string filename) = 0;
    virtual void readCommandsFromFile(std::string filename) = 0;
    virtual void redirectTrace(std::string trace, std::string dest, std::string filename) = 0;
    virtual void closeModel() = 0;
    virtual void createModel() = 0;
    virtual void execLinuxCommand(std::string command) = 0;
    virtual void verboseMode(bool on) = 0;
    virtual void check() = 0;
    virtual void getGenesysInfo() = 0;
    virtual void getCommandLine() = 0;
    virtual void sendFile(std::string filename, std::string hostname, std::string portname) = 0;
    virtual void setActivationCode(std::string code) = 0;
    virtual void receiveFile(std::string filename) = 0;
    virtual void startSimulation() = 0;
    virtual void stepSimulation() = 0;
    virtual void stopSimulation() = 0;
    virtual void showInit() = 0;
    virtual void showHelp() = 0;
    virtual void showHostName() = 0;
    //virtual void execute() = 0;
    //property TraceHandler : TProcTrace write aTraceHandler;
};

#endif /* GENESYSSHELL_IF_H */
```

8.27 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/applications/terminal/GenesysTerminalApp.cpp File Reference

```
#include "GenesysTerminalApp.h"
#include "../../kernel/simulator/Simulator.h"
#include <regex>
#include <fstream>
#include <assert.h>
```

Include dependency graph for GenesysTerminalApp.cpp:

8.28 GenesysTerminalApp.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   GenesysShell.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 23 de Maio de 2019, 13:02
 */

#include "GenesysTerminalApp.h"
#include "../../kernel/simulator/Simulator.h"
#include <regex>
#include <fstream>
#include <assert.h>

//#include "ProbDistribDefaultImpl1.h"

GenesysTerminalApp::GenesysTerminalApp() {
}

// AUX METHODS
 ----------------------------------------------------------------------------------------------------

void GenesysTerminalApp::Trace(std::string message) {
    std::cout « message « std::endl;
}

void GenesysTerminalApp::run(List<std::string>* commandlineArgs) {
    Trace("Genesys Shell is running. Type your command. For help, type the command \"help\".");
    //this->insertFakePluginsByHand(_simulator);
    std::string inputText; //,longPrefix, separator;
    while (true) {
        if (!commandlineArgs->empty()) {
            inputText = commandlineArgs->front();
            commandlineArgs->pop_front();
            tryExecuteCommand(inputText, "--", "=");
        } else {
```

```
            std::cout « _prompt « " ";
            std::getline(std::cin, inputText);
            tryExecuteCommand(inputText, "", " ");
        }
    }
}

void GenesysTerminalApp::tryExecuteCommand(std::string inputText, std::string longPrefix, std::string separator) {
    std::regex regex{R"([\s]+)"}; // split on space R"([\s]+)"
    std::sregex_token_iterator it{inputText.begin(), inputText.end(), regex, -1};
    std::vector<std::string> fields{it,{}};
    std::string typedCommandStr = fields[0];
    if (fields.size() > 1) { // its a command and a paramater in the form command parameter
        assert(fields.size() == 2);
        _parameter = fields[1];
    } else {
        _parameter = "";
    }
    ShellCommand* command;
    bool found;
    std::transform(typedCommandStr.begin(), typedCommandStr.end(), typedCommandStr.begin(), ::tolower);
    if (typedCommandStr.substr(0, 1) != "#") {
        found = true;
        if (typedCommandStr == "help") {
            std::string message = "";
            message += "command \tparameter \tdescription\n";
            message += "help    \t          \tShow this list\n";
            message += "exit    \t          \tQuit the terminal (same as exit)\n";
            message += "quit    \t          \tQuit the terminal (same as quit)\n";
            Trace(message);
        } else if (typedCommandStr == "exit" || typedCommandStr == "quit") {
            this->~GenesysTerminalApp();
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else if (typedCommandStr == "") {
        } else {
            found = false;
        }
        if (!found) {
            Trace("Command \"" + typedCommandStr + "\" not found. Type \"h\" or \"help\" for help.");
        }
    }

}

int GenesysTerminalApp::main(int argc, char** argv) {
    List<std::string>* commandlineArgs = new List<std::string>();
    for (unsigned short i = 1; i < argc; i++) {
        std::string arg = argv[i];
        commandlineArgs->insert(arg);
    }
    this->run(commandlineArgs);
    return 0;
}
```

8.29 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/applications/terminal/GenesysTerminal↩ App.h File Reference

```
#include "../BaseConsoleGenesysApplication.h"
#include "../../kernel/simulator/Simulator.h"
```

`#include "../../kernel/util/List.h"`

Include dependency graph for GenesysTerminalApp.h:

This graph shows which files directly or indirectly include this file:

Classes

- class GenesysTerminalApp

8.30 GenesysTerminalApp.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   GenesysTerminalApp.h
 * Author: rafael.luiz.cancian
 *
 * Created on 23 de Maio de 2019, 13:02
 */

#ifndef GENESYSTERMINALAPP_H
#define GENESYSTERMINALAPP_H

#include "../BaseConsoleGenesysApplication.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/util/List.h"

class GenesysTerminalApp : public BaseConsoleGenesysApplication {

    class ShellCommand {
    public:
        ShellCommand(std::string longname, std::string parameters, std::string descrition, unsigned short numParameters = 0) {
            this->descrition = descrition;
            this->longname = longname;
            this->parameters = parameters;
            this->numParameters = numParameters;
```

```
        }
        std::string longname;
        std::string parameters;
        std::string descrition;
        unsigned short numParameters;
    };
public:
    GenesysTerminalApp();
    virtual ~GenesysTerminalApp() = default;
public:
    virtual int main(int argc, char** argv);
private:
    void run(List<std::string>* arguments);
    void Trace(std::string message);
    void tryExecuteCommand(std::string inputText, std::string longPrefix, std::string separator);
private:
    Simulator* _simulator = new Simulator();
    std::string _prompt = "$Genesys> ";
    std::string _parameter;
};

#endif /* GENESYSTERMINALAPP_H */

```

8.31 html/dynsections.js File Reference

8.32 dynsections.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function toggleVisibility(linkObj)
{
 var base = $(linkObj).attr('id');
 var summary = $('#'+base+'-summary');
 var content = $('#'+base+'-content');
 var trigger = $('#'+base+'-trigger');
 var src=$(trigger).attr('src');
 if (content.is(':visible')===true) {
   content.hide();
   summary.show();
   $(linkObj).addClass('closed').removeClass('opened');
   $(trigger).attr('src',src.substring(0,src.length-8)+'closed.png');
 } else {
   content.show();
   summary.hide();
   $(linkObj).removeClass('closed').addClass('opened');
   $(trigger).attr('src',src.substring(0,src.length-10)+'open.png');
 }
 return false;
}

function updateStripes()
{
  $('table.directory tr').
       removeClass('even').filter(':visible:even').addClass('even');
}

function toggleLevel(level)
{
  $('table.directory tr').each(function() {
```

```
    var l = this.id.split('_').length-1;
    var i = $('#img'+this.id.substring(3));
    var a = $('#arr'+this.id.substring(3));
    if (l<level+1) {
      i.removeClass('iconfopen iconfclosed').addClass('iconfopen');
      a.html('&#9660;');
      $(this).show();
    } else if (l==level+1) {
      i.removeClass('iconfclosed iconfopen').addClass('iconfclosed');
      a.html('&#9658;');
      $(this).show();
    } else {
      $(this).hide();
    }
  });
  updateStripes();
}

function toggleFolder(id)
{
  // the clicked row
  var currentRow = $('#row_'+id);

  // all rows after the clicked row
  var rows = currentRow.nextAll("tr");

  var re = new RegExp('^row_'+id+'\\d+_$', "i"); //only one sub

  // only match elements AFTER this one (can't hide elements before)
  var childRows = rows.filter(function() { return this.id.match(re); });

  // first row is visible we are HIDING
  if (childRows.filter(':first').is(':visible')===true) {
    // replace down arrow by right arrow for current row
    var currentRowSpans = currentRow.find("span");
    currentRowSpans.filter(".iconfopen").removeClass("iconfopen").addClass("iconfclosed");
    currentRowSpans.filter(".arrow").html('&#9658;');
    rows.filter("[id^=row_"+id+"]").hide(); // hide all children
  } else { // we are SHOWING
    // replace right arrow by down arrow for current row
    var currentRowSpans = currentRow.find("span");
    currentRowSpans.filter(".iconfclosed").removeClass("iconfclosed").addClass("iconfopen");
    currentRowSpans.filter(".arrow").html('&#9660;');
    // replace down arrows by right arrows for child rows
    var childRowsSpans = childRows.find("span");
    childRowsSpans.filter(".iconfopen").removeClass("iconfopen").addClass("iconfclosed");
    childRowsSpans.filter(".arrow").html('&#9658;');
    childRows.show(); //show all children
  }
  updateStripes();
}


function toggleInherit(id)
{
  var rows = $('tr.inherit.'+id);
  var img = $('tr.inherit_header.'+id+' img');
  var src = $(img).attr('src');
  if (rows.filter(':first').is(':visible')===true) {
    rows.css('display','none');
    $(img).attr('src',src.substring(0,src.length-8)+'closed.png');
  } else {
    rows.css('display','table-row'); // using show() causes jump in firefox
    $(img).attr('src',src.substring(0,src.length-10)+'open.png');
  }
}
/* @license-end */
```

8.33 user/html/dynsections.js File Reference

8.34 dynsections.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
```

```
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function toggleVisibility(linkObj)
{
 var base = $(linkObj).attr('id');
 var summary = $('#'+base+'-summary');
 var content = $('#'+base+'-content');
 var trigger = $('#'+base+'-trigger');
 var src=$(trigger).attr('src');
 if (content.is(':visible')===true) {
   content.hide();
   summary.show();
   $(linkObj).addClass('closed').removeClass('opened');
   $(trigger).attr('src',src.substring(0,src.length-8)+'closed.png');
 } else {
   content.show();
   summary.hide();
   $(linkObj).removeClass('closed').addClass('opened');
   $(trigger).attr('src',src.substring(0,src.length-10)+'open.png');
 }
 return false;
}

function updateStripes()
{
  $('table.directory tr').
       removeClass('even').filter(':visible:even').addClass('even');
}

function toggleLevel(level)
{
  $('table.directory tr').each(function() {
    var l = this.id.split('_').length-1;
    var i = $('#img'+this.id.substring(3));
    var a = $('#arr'+this.id.substring(3));
    if (l<level+1) {
      i.removeClass('iconfopen iconfclosed').addClass('iconfopen');
      a.html('&#9660;');
      $(this).show();
    } else if (l==level+1) {
      i.removeClass('iconfclosed iconfopen').addClass('iconfclosed');
      a.html('&#9658;');
      $(this).show();
    } else {
      $(this).hide();
    }
  });
  updateStripes();
}

function toggleFolder(id)
{
  // the clicked row
  var currentRow = $('#row_'+id);

  // all rows after the clicked row
  var rows = currentRow.nextAll("tr");

  var re = new RegExp('^row_'+id+'\\d+_$', "i"); //only one sub

  // only match elements AFTER this one (can't hide elements before)
  var childRows = rows.filter(function() { return this.id.match(re); });

  // first row is visible we are HIDING
  if (childRows.filter(':first').is(':visible')===true) {
    // replace down arrow by right arrow for current row
    var currentRowSpans = currentRow.find("span");
    currentRowSpans.filter(".iconfopen").removeClass("iconfopen").addClass("iconfclosed");
    currentRowSpans.filter(".arrow").html('&#9658;');
    rows.filter("[id^=row_"+id+"]").hide(); // hide all children
  } else { // we are SHOWING
    // replace right arrow by down arrow for current row
    var currentRowSpans = currentRow.find("span");
    currentRowSpans.filter(".iconfclosed").removeClass("iconfclosed").addClass("iconfopen");
    currentRowSpans.filter(".arrow").html('&#9660;');
```

```
    // replace down arrows by right arrows for child rows
    var childRowsSpans = childRows.find("span");
    childRowsSpans.filter(".iconfopen").removeClass("iconfopen").addClass("iconfclosed");
    childRowsSpans.filter(".arrow").html('&#9658;');
    childRows.show(); //show all children
  }
  updateStripes();
}


function toggleInherit(id)
{
  var rows = $('tr.inherit.'+id);
  var img = $('tr.inherit_header.'+id+' img');
  var src = $(img).attr('src');
  if (rows.filter(':first').is(':visible')===true) {
    rows.css('display','none');
    $(img).attr('src',src.substring(0,src.length-8)+'closed.png');
  } else {
    rows.css('display','table-row'); // using show() causes jump in firefox
    $(img).attr('src',src.substring(0,src.length-10)+'open.png');
  }
}
/* @license-end */

$(document).ready(function() {
  $('.code,.codeRef').each(function() {

    $(this).data('powertip',$('#a'+$(this).attr('href').replace(/.*\//,'').replace(/[^a-z_A-Z0-9]/g,'_')).html());
    $(this).powerTip({ placement: 's', smartPlacement: true, mouseOnToPopup: true });
  });
});
```

8.35 html/jquery.js File Reference

8.36 jquery.js

```
/*! jQuery v3.4.1 | (c) JS Foundation and other contributors | jquery.org/license */
!function(e,t){"use strict";"object"==typeof module&&"object"==typeof
module.exports?module.exports=e.document?t(e,!0):function(e){if(!e.document)throw new Error("jQuery
requires a window with a document");return t(e)}:t(e)}("undefined"!=typeof
window?window:this,function(C,e){"use strict";var
t=[],E=C.document,r=Object.getPrototypeOf,s=t.slice,g=t.concat,u=t.push,i=t.indexOf,n={},o=n.toString,v=n.hasOwnPropert
e&&"number"!=typeof e.nodeType},x=function(e){return
null!=e&&e===e.window},c={type:!0,src:!0,nonce:!0,noModule:!0};function b(e,t,n){var
r,i,o=(n=n||E).createElement("script");if(o.text=e,t)for(r in
c)(i=t[r]||t.getAttribute&&t.getAttribute(r))&&o.setAttribute(r,i);n.head.appendChild(o).parentNode.removeChild(o)}func
w(e){return null==e?e+"":"object"==typeof e||"function"==typeof e?n[o.call(e)]||"object":typeof e}var
f="3.4.1",k=function(e,t){return new k.fn.init(e,t)},p=/^[\s\uFEFF\xA0]+|[\s\uFEFF\xA0]+$/g;function
d(e){var t=!!e&&"length"in
e&&e.length,n=w(e);return!m(e)&&!x(e)&&("array"===n||0===t||"number"==typeof t&&0<t&&t-1 in
e)}k.fn=k.prototype={jquery:f,constructor:k,length:0,toArray:function(){return
s.call(this)},get:function(e){return
null==e?s.call(this):e<0?this[e+this.length]:this[e]},pushStack:function(e){var
t=k.merge(this.constructor(),e);return t.prevObject=this,t},each:function(e){return
k.each(this,e)},map:function(n){return this.pushStack(k.map(this,function(e,t){return
n.call(e,t,e)}))},slice:function(){return
this.pushStack(s.apply(this,arguments))},first:function(){return this.eq(0)},last:function(){return
this.eq(-1)},eq:function(e){var t=this.length,n=+e+(e<0?t:0);return
this.pushStack(0<=n&&n<t?[this[n]]:[])},end:function(){return
this.prevObject||this.constructor()},push:u,sort:t.sort,splice:t.splice},k.extend=k.fn.extend=function(){var
e,t,n,r,i,o,a=arguments[0]||{},s=1,u=arguments.length,l=!1;for("boolean"==typeof
a&&(l=a,a=arguments[s]||{},s++),"object"==typeof
a||m(a)||(a={}),s===u&&(a=this,s--);s<u;s++)if(null!=(e=arguments[s]))for(t in
e)r=e[t],"__proto__"!==t&&a!==r&&(l&&r&&(k.isPlainObject(r)||(i=Array.isArray(r)))?(n=a[t],o=i&&!Array.isArray(n)?[]:i
0!==r&&(a[t]=r));return
a},k.extend({expando:"jQuery"+(f+Math.random()).replace(/\D/g,""),isReady:!0,error:function(e){throw
new Error(e)},noop:function(){},isPlainObject:function(e){var t,n;return!(!e||"[object
Object]"!==o.call(e))&&(!(t=r(e))||"function"==typeof(n=v.call(t,"constructor")&&t.constructor)&&a.call(n)===l)},isEmpt
t;for(t in
e)return!1;return!0},globalEval:function(e,t){b(e,{nonce:t&&t.nonce})},each:function(e,t){var
n,r=0;if(d(e)){for(n=e.length;r<n;r++)if(!1===t.call(e[r],r,e[r]))break}else for(r in
e)if(!1===t.call(e[r],r,e[r]))break;return e},trim:function(e){return
null==e?"":(e+"").replace(p,"")},makeArray:function(e,t){var n=t||[];return
null!=e&&(d(Object(e))?k.merge(n,"string"==typeof
e?[e]:e):u.call(n,e)),n},inArray:function(e,t,n){return
null==t?-1:i.call(t,e,n)},merge:function(e,t){for(var
n=+t.length,r=0,i=e.length;r<n;r++)e[i++]=t[r];return e.length=i,e},grep:function(e,t,n){for(var
r=[],i=0,o=e.length,a=!n;i<o;i++)!t(e[i],i)!==a&&r.push(e[i]);return r},map:function(e,t,n){var
r,i,o=0,a=[];if(d(e))for(r=e.length;o<r;o++)null!=(i=t(e[o],o,n))&&a.push(i);else for(o in
e)null!=(i=t(e[o],o,n))&&a.push(i);return g.apply([],a)},guid:1,support:y}),"function"==typeof
Symbol&&(k.fn[Symbol.iterator]=t[Symbol.iterator]),k.each("Boolean Number String Function Array Date
```

```
RegExp Object Error Symbol".split(" "),function(e,t){n["[object "+t+"]"]=t.toLowerCase()});var
h=function(n){var e,d,b,o,i,h,f,g,w,u,l,T,C,a,E,v,s,c,y,k="sizzle"+1*new
Date,m=n.document,S=0,r=0,p=ue(),x=ue(),N=ue(),A=ue(),D=function(e,t){return
e===t&&(l=!0),0},j={}.hasOwnProperty,t=[],q=t.pop,L=t.push,H=t.push,O=t.slice,P=function(e,t){for(var
n=0,r=e.length;n<r;n++)if(e[n]===t)return
n;return-1},R="checked|selected|async|autofocus|autoplay|controls|defer|disabled|hidden|ismap|loop|multiple|open|readon
RegExp(M+"+","g"),B=new RegExp("^"+M+"+|((?:^|[^\\\\])(?:\\\\.)*)"+M+"+$","g"),_=new
RegExp("^"+M+"*,"+M+"*"),z=new RegExp("^"+M+"*([>+~]|"+M+")"+M+"*"),U=new RegExp(M+"|>"),X=new
RegExp($),V=new RegExp("^"+I+"$"),G={ID:new RegExp("^#("+I+")"),CLASS:new
RegExp("^\\.("+I+")"),TAG:new RegExp("^("+I+"|[*])"),ATTR:new RegExp("^"+W),PSEUDO:new
RegExp("^"+$),CHILD:new
RegExp("^:(only|first|last|nth|nth-last)-(child|of-type)(?:\\("+M+"*(even|odd|(([+-]|)(\\d*)n|)"+M+"*(?:([+-]|)"+M+"*(\
RegExp("^(?:"+R+")$","i"),needsContext:new
RegExp("^"+M+"*[>+~]|:(even|odd|eq|gt|lt|nth|first|last)(?:\\("+M+"*((?:-\\d)?\\d*)"+M+"*\\)|)(?=[^-]|$)","i")},Y=/HTMI
\w/,Z=/^(?:#([\w-]+)|(\w+)|\.([\w-]+))$/,ee=/[+~]/,te=new
RegExp("\\\\([\\da-f]{1,6}"+M+"?|("+M+")|.)","ig"),ne=function(e,t,n){var r="0x"+t-65536;return
r!=r||n?t:r<0?String.fromCharCode(r+65536):String.fromCharCode(r»10|55296,1023&r|56320)},re=/([\0-\x1f\x7f]|^-?\d)|^-$|
t?"\0"===e?"\ufffd":e.slice(0,-1)+"\\"+e.charCodeAt(e.length-1).toString(16)+"
":"\\"+e},oe=function(){T()},ae=be(function(e){return!0===e.disabled&&"fieldset"===e.nodeName.toLowerCase()},{dir:"pare
n=e.length,r=0;while(e[n++]=t[r++]);e.length=n-1}}}function se(t,e,n,r){var
i,o,a,s,u,l,c,f=e&&e.ownerDocument,p=e?e.nodeType:9;if(n=n||[],"string"!=typeof
t||!t||1!==p&&9!==p&&11!==p)return
n;if(!r&&((e?e.ownerDocument||e:m)!==C&&T(e),e=e||C,E)){if(11!==p&&(u=Z.exec(t)))if(i=u[1]){if(9===p){if(!(a=e.getEleme
n;if(a.id===i)return n.push(a),n}else if(f&&(a=f.getElementById(i))&&y(e,a)&&a.id===i)return
n.push(a),n}else{if(u[2])return
H.apply(n,e.getElementsByTagName(t)),n;if((i=u[3])&&d.getElementsByClassName&&e.getElementsByClassName)return
H.apply(n,e.getElementsByClassName(i)),n}if(d.qsa&&!A[t+"
"]&&(!v||!v.test(t))&&(1!==p||"object"!==e.nodeName.toLowerCase())){if(c=t,f=e,1===p&&U.test(t)){(s=e.getAttribute("id'
"+xe(l[o]);c=l.join(","),f=ee.test(t)&&ye(e.parentNode)||e}try{return
H.apply(n,f.querySelectorAll(c)),n}catch(e){A(t,!0)}finally{s===k&&e.removeAttribute("id")}}}return
g(t.replace(B,"$1"),e,n,r)}function ue(){var r=[];return function e(t,n){return r.push(t+"
")>b.cacheLength&&delete e[r.shift()],e[t+" "]=n}}function le(e){return e[k]=!0,e}function ce(e){var
t=C.createElement("fieldset");try{return!!e(t)}catch(e){return!1}finally{t.parentNode&&t.parentNode.removeChild(t),t=nu
fe(e,t){var n=e.split("|"),r=n.length;while(r--)b.attrHandle[n[r]]=t}function pe(e,t){var
n=t&&e,r=n&&1===e.nodeType&&1===t.nodeType&&e.sourceIndex-t.sourceIndex;if(r)return
r;if(n)while(n=n.nextSibling)if(n===t)return-1;return e?1:-1}function de(t){return
function(e){return"input"===e.nodeName.toLowerCase()&&e.type===t}}function he(n){return
function(e){var t=e.nodeName.toLowerCase();return("input"===t||"button"===t)&&e.type===n}}function
ge(t){return function(e){return"form"in e?e.parentNode&&!1===e.disabled?"label"in e?"label"in
e.parentNode?e.parentNode.disabled===t:e.disabled===t:e.isDisabled===t||e.isDisabled!==!t&&ae(e)===t:e.disabled===t:"la
e&&e.disabled===t}}function ve(a){return le(function(o){return o=+o,le(function(e,t){var
n,r=a([],e.length,o),i=r.length;while(i--)e[n=r[i]]&&(e[n]=!(t[n]=e[n]))})})}function ye(e){return
e&&"undefined"!=typeof e.getElementsByTagName&&e}for(e in d=se.support={},i=se.isXML=function(e){var
t=e.namespaceURI,n=(e.ownerDocument||e).documentElement;return!Y.test(t||n&&n.nodeName||"HTML")},T=se.setDocument=funct
t,n,r=e?e.ownerDocument||e:m;return
r!==C&&9===r.nodeType&&r.documentElement&&(a=(C=r).documentElement,E=!i(C),m!==C&&(n=C.defaultView)&&n.top!==n&&(n.addE
e.className="i",!e.getAttribute("className")}),d.getElementsByTagName=ce(function(e){return
e.appendChild(C.createComment("")),!e.getElementsByTagName("*").length}),d.getElementsByClassName=K.test(C.getElementsE
a.appendChild(e).id=k,!C.getElementsByName||!C.getElementsByName(k).length}),d.getById?(b.filter.ID=function(e){var
t=e.replace(te,ne);return function(e){return
e.getAttribute("id")===t}},b.find.ID=function(e,t){if("undefined"!=typeof t.getElementById&&E){var
n=t.getElementById(e);return n?[n]:[]}}):(b.filter.ID=function(e){var n=e.replace(te,ne);return
function(e){var t="undefined"!=typeof e.getAttributeNode&&e.getAttributeNode("id");return
t&&t.value===n}},b.find.ID=function(e,t){if("undefined"!=typeof t.getElementById&&E){var
n,r,i,o=t.getElementById(e);if(o){if((n=o.getAttributeNode("id"))&&n.value===e)return[o];i=t.getElementsByName(e),r=0;w
t.getElementsByTagName?t.getElementsByTagName(e):d.qsa?t.querySelectorAll(e):void
0}:function(e,t){var
n,r=[],i=0,o=t.getElementsByTagName(e);if("*"===e){while(n=o[i++])1===n.nodeType&&r.push(n);return
r}return o},b.find.CLASS=d.getElementsByClassName&&function(e,t){if("undefined"!=typeof
t.getElementsByClassName&&E)return
t.getElementsByClassName(e)},s=[],v=[],(d.qsa=K.test(C.querySelectorAll))&&(ce(function(e){a.appendChild(e).innerHTML="
id='"+k+"'></a><select id='"+k+"-\r\' msallowcapture=''><option
selected=''></option></select>",e.querySelectorAll("[msallowcapture^='']").length&&v.push("[*^$]="+M+"*(?:''|\"\")"),e.que
href='' disabled='disabled'></a><select disabled='disabled'><option/></select>";var
t=C.createElement("input");t.setAttribute("type","hidden"),e.appendChild(t).setAttribute("name","D"),e.querySelectorAll
RegExp(v.join("|")),s=s.length&&new
RegExp(s.join("|")),t=K.test(a.compareDocumentPosition),y=t||K.test(a.contains)?function(e,t){var
n=9===e.nodeType?e.documentElement:e,r=t&&t.parentNode;return
e===r||!(!r||1!==r.nodeType||!(n.contains?n.contains(r):e.compareDocumentPosition&&16&e.compareDocumentPosition(r)))}:
l=!0,0;var n=!e.compareDocumentPosition-!t.compareDocumentPosition;return
n||(1&(n=(e.ownerDocument||e)===(t.ownerDocument||t)?e.compareDocumentPosition(t):1)||!d.sortDetached&&t.compareDocumen
l=!0,0;var n,r=0,i=e.parentNode,o=t.parentNode,a=[e],s=[t];if(!i||!o)return
e===C?-1:t===C?1:i?-1:o?1:u?P(u,e)-P(u,t):0;if(i===o)return
pe(e,t);n=e;while(n=n.parentNode)a.unshift(n);n=t;while(n=n.parentNode)s.unshift(n);while(a[r]===s[r])r++;return
r?pe(a[r],s[r]):a[r]===m?-1:s[r]===m?1:0})},C},se.matches=function(e,t){return
se(e,null,null,t)},se.matchesSelector=function(e,t){if((e.ownerDocument||e)!==C&&T(e),d.matchesSelector&&E&&!A[t+"
"]&&(!s||!s.test(t))&&(!v||!v.test(t)))try{var
n=c.call(e,t);if(n||d.disconnectedMatch||e.document&&11!==e.document.nodeType)return
n}catch(e){A(t,!0)}return
0<se(t,C,null,[e]).length},se.contains=function(e,t){return(e.ownerDocument||e)!==C&&T(e),y(e,t)},se.attr=function(e,t)
n=b.attrHandle[t.toLowerCase()],r=n&&j.call(b.attrHandle,t.toLowerCase())?n(e,t,!E):void 0;return
void
0!==r?r:d.attributes||!E?e.getAttribute(t):(r=e.getAttributeNode(t))&&r.specified?r.value:null},se.escape=function(e){r
new Error("Syntax error, unrecognized expression: "+e)},se.uniqueSort=function(e){var
t,n=[],r=0,i=0;if(l=!d.detectDuplicates,u=!d.sortStable&&e.slice(0),e.sort(D),l){while(t=e[i++])t===e[i]&&(r=n.push(i)
u=null,e},o=se.getText=function(e){var
t,n="",r=0,i=e.nodeType;if(i){if(1===i||9===i||11===i){if("string"==typeof e.textContent)return
```

```
e.textContent;for(e=e.firstChild;e;e=e.nextSibling)n+=o(e)}else if(3===i||4===i)return
e.nodeValue}else while(t=e[r++])n+=o(t);return
n},(b=se.selectors={cacheLength:50,createPseudo:le,match:G,attrHandle:{},find:{},relative:{">":{dir:"parentNode",first
":{dir:"parentNode"},"+":{dir:"previousSibling",first:!0},"~":{dir:"previousSibling"}},preFilter:{ATTR:function(e){retu
e[1]=e[1].replace(te,ne),e[3]=(e[3]||e[4]||e[5]||"").replace(te,ne),"~="===e[2]&&(e[3]=" "+e[3]+"
"),e.slice(0,4)},CHILD:function(e){return
e[1]=e[1].toLowerCase(),"nth"===e[1].slice(0,3)?(e[3]||se.error(e[0]),e[4]=+(e[4]?e[5]+(e[6]||1):2*("even"===e[3]||"odd
t,n=!e[6]&&e[2];return
G.CHILD.test(e[0])?null:(e[3]?e[2]=e[4]||e[5]||"":n&&X.test(n)&&(t=h(n,!0))&&(t=n.indexOf(")",n.length-t)-n.length)&&(e
t=e.replace(te,ne).toLowerCase();return"*"===e?function(){return!0}:function(e){return
e.nodeName&&e.nodeName.toLowerCase()===t}},CLASS:function(e){var t=p[e+" "];return t||(t=new
RegExp("(^|"+M+")"+e+"("+M+"|$)"))&&p(e,function(e){return t.test("string"==typeof
e.className&&e.className||"undefined"!=typeof
e.getAttribute&&e.getAttribute("class")||"")})},ATTR:function(n,r,i){return function(e){var
t=se.attr(e,n);return
null==t?"!="===r:!r||(t+="","="===r?t===i:"!="===r?t!==i:"^="===r?i&&0===t.indexOf(i):"*="===r?i&&-1<t.indexOf(i):"$="
"+t.replace(F," ")+"
").indexOf(i):"|="===r&&(t===i||t.slice(0,i.length+1)===i+"-"))}},CHILD:function(h,e,t,g,v){var
y="nth"!==h.slice(0,3),m="last"!==h.slice(-4),x="of-type"===e;return
1===g&&0===v?function(e){return!!e.parentNode}:function(e,t,n){var
r,i,o,a,s,u,l=y!==m?"nextSibling":"previousSibling",c=e.parentNode,f=x&&e.nodeName.toLowerCase(),p=!n&&!x,d=!1;if(c){i
if(p&&(d=s=(r=(i=(o=(a=e)[k]||(a[k]={}))[a.uniqueID]||(o[a.uniqueID]={}))[h]||[])[0]===S&&r[1]),!1===d)while(a=++s&&a&&
t,a=b.pseudos[e]||b.setFilters[e.toLowerCase()]||se.error("unsupported pseudo: "+e);return
a[k]?a(o):1<a.length?(t=[e,e,"",o],b.setFilters.hasOwnProperty(e.toLowerCase())?le(function(e,t){var
n,r=a(e,o),i=r.length;while(i--)e[n=P(e,r[i])]=!(t[n]=r[i])}):function(e){return
a(e,0,t)}):a}},pseudos:{not:le(function(e){var r=[],i=[],s=f(e.replace(B,"$1"));return
s[k]?le(function(e,t,n,r){var
i,o=s(e,null,r,[]),a=e.length;while(a--)(i=o[a])&&(e[a]=!(t[a]=i))}):function(e,t,n){return
r[0]=e,s(r,null,n,i),r[0]=null,!i.pop()}}),has:le(function(t){return function(e){return
0<se(t,e).length}}),contains:le(function(t){return
t=t.replace(te,ne),function(e){return-1<(e.textContent||o(e)).indexOf(t)}}),lang:le(function(n){return
V.test(n||"")||se.error("unsupported lang: "+n),n=n.replace(te,ne).toLowerCase(),function(e){var
t;do{if(t=E?e.lang:e.getAttribute("xml:lang")||e.getAttribute("lang"))return(t=t.toLowerCase())===n||0===t.indexOf(n+"
t=n.location&&n.location.hash;return t&&t.slice(1)===e.id},root:function(e){return
e===a},focus:function(e){return
e===C.activeElement&&(!C.hasFocus||C.hasFocus())&&!!(e.type||e.href||~e.tabIndex)},enabled:ge(!1),disabled:ge(!0),check
t=e.nodeName.toLowerCase();return"input"===t&&!!e.checked||"option"===t&&!!e.selected},selected:function(e){return
e.parentNode&&e.parentNode.selectedIndex,!0===e.selected},empty:function(e){for(e=e.firstChild;e;e=e.nextSibling)if(e.n
J.test(e.nodeName)},input:function(e){return Q.test(e.nodeName)},button:function(e){var
t=e.nodeName.toLowerCase();return"input"===t&&"button"===e.type||"button"===t},text:function(e){var
t;return"input"===e.nodeName.toLowerCase()&&"text"===e.type&&(null==(t=e.getAttribute("type"))||"text"===t.toLowerCase
n=0;n<t;n+=2)e.push(n);return e}),odd:ve(function(e,t){for(var n=1;n<t;n+=2)e.push(n);return
e}),lt:ve(function(e,t,n){for(var r=n<0?n+t:t<n?t:n;0<=--r;)e.push(r);return
e}),gt:ve(function(e,t,n){for(var r=n<0?n+t:n;++r<t;)e.push(r);return
e})}}).pseudos.nth=b.pseudos.eq,{radio:!0,checkbox:!0,file:!0,password:!0,image:!0})b.pseudos[e]=de(e);for(e
in{submit:!0,reset:!0})b.pseudos[e]=he(e);function me(){}function xe(e){for(var
t=0,n=e.length,r="";t<n;t++)r+=e[t].value;return r}function be(s,e,t){var
u=e.dir,l=e.next,c=l||u,f=t&&"parentNode"===c,p=r++;return
e.first?function(e,t,n){while(e=e[u])if(1===e.nodeType||f)return
s(e,t,n);return!1}:function(e,t,n){var
r,i,o,a=[S,p];if(n){while(e=e[u])if((1===e.nodeType||f)&&s(e,t,n))return!0}else
while(e=e[u])if(1===e.nodeType||f)if(i=(o=e[k]||(e[k]={}))[e.uniqueID]||(o[e.uniqueID]={}),l&&l===e.nodeName.toLowerCas
a[2]=r[2];if((i[c]=a)[2]=s(e,t,n))return!0}return!1}}function we(i){return
1<i.length?function(e,t,n){var r=i.length;while(r--)if(!i[r](e,t,n))return!1;return!0}:i[0]}function
Te(e,t,n,r,i){for(var
o,a=[],s=0,u=e.length,l=null!=t;s<u;s++)(o=e[s])&&(n&&!n(o,r,i)||(a.push(o),l&&t.push(s)));return
a}function Ce(d,h,g,v,y,e){return v&&!v[k]&&(v=Ce(v)),y&&!y[k]&&(y=Ce(y,e)),le(function(e,t,n,r){var
i,o,a,s=[],u=[],l=t.length,c=e||function(e,t,n){for(var r=0,i=t.length;r<i;r++)se(e,t[r],n);return
n}(h||"*",n.nodeType?[n]:n,[]),f=!d||!e&&h?c:Te(c,s,d,n,r),p=g?y||(e?d:l||v)?[]:t:f;if(g&&g(f,p,n,r),v){i=Te(p,u),v(i,
p=Te(p===t?p.splice(l,p.length):p),y?y(null,t,p,r):H.apply(t,p)})}function Ee(e){for(var
i,t,n,r=e.length,o=b.relative[e[0].type],a=o||b.relative[" "],s=o?1:0,u=be(function(e){return
e===i},a,!0),l=be(function(e){return-1<P(i,e)},a,!0),c=[function(e,t,n){var
r=!o&&(n||t!==w)||((i=t).nodeType?u(e,t,n):l(e,t,n));return
i=null,r}];s<r;s++)if(t=b.relative[e[s].type])c=[be(we(c),t)];else{if((t=b.filter[e[s].type].apply(null,e[s].matches))
Ce(1<s&&we(c),1<s&&xe(e.slice(0,s-1).concat({value:"
"===e[s-2].type?"*":""})).replace(B,"$1"),t,s<n&&Ee(e.slice(s,n)),n<r&&Ee(e=e.slice(n)),n<r&&xe(e))}c.push(t)}return
we(c)}return me.prototype=b.filters=b.pseudos,b.setFilters=new me,h=se.tokenize=function(e,t){var
n,r,i,o,a,s,u,l=x[e+" "];if(l)return t?0:l.slice(0);a=e,s=[],u=b.preFilter;while(a){for(o in
n&&!(r=_.exec(a))||(r&&(a=a.slice(r[0].length)||a),s.push(i=[])),n=!1,(r=z.exec(a))&&(n=r.shift(),i.push({value:n,type
")}),a=a.slice(n.length)),b.filter)!(r=G[o].exec(a))||u[o]&&!(r=u[o](r))||(n=r.shift(),i.push({value:n,type:o,matches:r
t?a.length:a?se.error(e):x(e,s).slice(0)},f=se.compile=function(e,t){var
n,v,y,m,x,r,i=[],o=[],a=N[e+"
"];if(!a){t||(t=h(e)),n=t.length;while(n--)(a=Ee(t[n]))[k]?i.push(a):o.push(a);(a=N(e,(v=o,m=0<(y=i).length,x=0<v.lengt
o,a,s,u=0,l="0",c=e&&[],f=[],p=w,d=e||x&&b.find.TAG("*",i),h=S+=null==p?1:Math.random()||.1,g=d.length;for(i&&(w=t===C
i&&(S=h,w=p),c},m?le(r):r)).selector=e}return a},g=se.select=function(e,t,n,r){var
i,o,a,s,u,l="function"==typeof
e&&e,c=!r&&h(e=l.selector||e);if(n=n||[],1===c.length){if(2<(o=c[0]=c[0].slice(0)).length&&"ID"===(a=o[0]).type&&9===t
n;l&&(t=t.parentNode),e=e.slice(o.shift().value.length)}i=G.needsContext.test(e)?0:o.length;while(i--){if(a=o[i],b.rela
H.apply(n,r),n;break}}}return(l||f(e,c))(r,t,!E,n,!t||ee.test(e)&&ye(t.parentNode)||t),n},d.sortStable=k.split("").sort
1&e.compareDocumentPosition(C.createElement("fieldset"))}),ce(function(e){return e.innerHTML="<a
href='#'></a>","#"===e.firstChild.getAttribute("href")})||fe("type|href|height|width",function(e,t,n){if(!n)return
e.getAttribute(t,"type"===t.toLowerCase()?1:2)}),d.attributes&&ce(function(e){return
e.innerHTML="<input/>",e.firstChild.setAttribute("value",""),""===e.firstChild.getAttribute("value")})||fe("value",func
e.defaultValue}),ce(function(e){return null==e.getAttribute("disabled")})||fe(R,function(e,t,n){var
r;if(!n)return!0===e[t]?t.toLowerCase():(r=e.getAttributeNode(t))&&r.specified?r.value:null}),se}(C);k.find=h,k.expr=h
T=function(e,t,n){var r=[],i=void
0!==n;while((e=e[t])&&9!==e.nodeType)if(1===e.nodeType){if(i&&k(e).is(n))break;r.push(e)}return
```

```
r},S=function(e,t){for(var n=[];e;e=e.nextSibling)1===e.nodeType&&e!==t&&n.push(e);return
n},N=k.expr.match.needsContext;function A(e,t){return
e.nodeName&&e.nodeName.toLowerCase()===t.toLowerCase()}var
D=/^<([a-z][^\/\0>:\x20\t\r\n\f]*)[\x20\t\r\n\f]*\/?>(?:<\/\1>|)$/i;function j(e,n,r){return
m(n)?k.grep(e,function(e,t){return!!n.call(e,t,e)!==r}):n.nodeType?k.grep(e,function(e){return
e===n!==r}):"string"!=typeof
n?k.grep(e,function(e){return-1<i.call(n,e)!==r}):k.filter(n,e,r)}k.filter=function(e,t,n){var
r=t[0];return
n&&(e=":not("+e+")"),1===t.length&&1===r.nodeType?k.find.matchesSelector(r,e)?[r]:[]:k.find.matches(e,k.grep(t,function
1===e.nodeType}))},k.fn.extend({find:function(e){var t,n,r=this.length,i=this;if("string"!=typeof
e)return
this.pushStack(k(e).filter(function(){for(t=0;t<r;t++)if(k.contains(i[t],this))return!0}));for(n=this.pushStack([]),t=0
1<r?k.uniqueSort(n):n},filter:function(e){return
this.pushStack(j(this,e||[],!1))},not:function(e){return
this.pushStack(j(this,e||[],!0))},is:function(e){return!!j(this,"string"==typeof
e&&N.test(e)?k(e):e||[],!1).length}});var
q,L=/^(?:\s*(<[\w\W]+>)[^>]*|#([\w-]+))$/;(k.fn.init=function(e,t,n){var r,i;if(!e)return
this;if(n=n||q,"string"==typeof
e){if(!(r="<"===e[0]&&">"===e[e.length-1]&&3<=e.length?[null,e,null]:L.exec(e))||!r[1]&&t)return!t||t.jquery?(t||n).fin
instanceof
k?t[0]:t,k.merge(this,k.parseHTML(r[1],t&&t.nodeType?t.ownerDocument||t:E,!0)),D.test(r[1])&&k.isPlainObject(t))for(r
in t)m(this[r])?this[r](t[r]):this.attr(r,t[r]);return
this}return(i=E.getElementById(r[2]))&&(this[0]=i,this.length=1),this}return
e.nodeType?(this[0]=e,this.length=1,this):m(e)?void
0!==n.ready?n.ready(e):e(k):k.makeArray(e,this)}).prototype=k.fn,q=k(E);var
H=/^(?:parents|prev(?:Until|All))/,O={children:!0,contents:!0,next:!0,prev:!0};function
P(e,t){while((e=e[t])&&1!==e.nodeType);return e}k.fn.extend({has:function(e){var
t=k(e,this),n=t.length;return this.filter(function(){for(var
e=0;e<n;e++)if(k.contains(this,t[e]))return!0})},closest:function(e,t){var
n,r=0,i=this.length,o=[],a="string"!=typeof
e&&k(e);if(!N.test(e))for(;r<i;r++)for(n=this[r];n&&n!==t;n=n.parentNode)if(n.nodeType<11&&(a?-1<a.index(n):1===n.node
this.pushStack(1<o.length?k.uniqueSort(o):o)},index:function(e){return e?"string"==typeof
e?i.call(k(e),this[0]):i.call(this,e.jquery?e[0]:e):this[0]&&this[0].parentNode?this.first().prevAll().length:-1},add:
this.pushStack(k.uniqueSort(k.merge(this.get(),k(e,t))))},addBack:function(e){return
this.add(null==e?this.prevObject:this.prevObject.filter(e))}}),k.each({parent:function(e){var
t=e.parentNode;return t&&11!==t.nodeType?t:null},parents:function(e){return
T(e,"parentNode")},parentsUntil:function(e,t,n){return T(e,"parentNode",n)},next:function(e){return
P(e,"nextSibling")},prev:function(e){return P(e,"previousSibling")},nextAll:function(e){return
T(e,"nextSibling")},prevAll:function(e){return
T(e,"previousSibling")},nextUntil:function(e,t,n){return
T(e,"nextSibling",n)},prevUntil:function(e,t,n){return
T(e,"previousSibling",n)},siblings:function(e){return
S((e.parentNode||{}).firstChild,e)},children:function(e){return
S(e.firstChild)},contents:function(e){return"undefined"!=typeof
e.contentDocument?e.contentDocument:(A(e,"template")&&(e=e.content||e),k.merge([],e.childNodes))}},function(r,i){k.fn[
n=k.map(this,i,e);return"Until"!==r.slice(-5)&&(t=e),t&&"string"==typeof
t&&(n=k.filter(t,n)),1<this.length&&(O[r]||k.uniqueSort(n),H.test(r)&&n.reverse()),this.pushStack(n)}});var
R=/[^\x20\t\r\n\f]+/g;function M(e){return e}function I(e){throw e}function W(e,t,n,r){var
i;try{e&&m(i=e.promise)?i.call(e).done(t).fail(n):e&&m(i=e.then)?i.call(e,t,n):t.apply(void
0,[e].slice(r))}catch(e){n.apply(void 0,[e])}}k.Callbacks=function(r){var e,n;r="string"==typeof
r?(e=r,n={},k.each(e.match(R)||[],function(e,t){n[t]=!0})),n):k.extend({},r);var
i,t,o,a,s=[],u=[],l=-1,c=function(){for(a=a||r.once,o=i=!0;u.length;l=-1){t=u.shift();while(++l<s.length)!1===s[l].appl
s&&(t&&!i&&(l=s.length-1,u.push(t)),function
n(e){k.each(e,function(e,t){m(t)?r.unique&&f.has(t)||s.push(t):t&&t.length&&"string"!==w(t)&&n(t)})}(arguments),t&&!i&
k.each(arguments,function(e,t){var
n;while(-1<(n=k.inArray(t,s,n)))s.splice(n,1),n<=l&&l--})},this},has:function(e){return
e?-1<k.inArray(e,s):0<s.length},empty:function(){return s&&(s=[]),this},disable:function(){return
a=u=[],s=t="",this},disabled:function(){return!s},lock:function(){return
a=u=[],t||i||(s=t=""),this},locked:function(){return!!a},fireWith:function(e,t){return
a||(t=[e,(t=t||[])].slice?t.slice():t],u.push(t),i||c()),this},fire:function(){return
f.fireWith(this,arguments),this},fired:function(){return!!o}};return
f},k.extend({Deferred:function(e){var
o=[["notify","progress",k.Callbacks("memory"),k.Callbacks("memory"),2],["resolve","done",k.Callbacks("once
memory"),k.Callbacks("once memory"),0,"resolved"],["reject","fail",k.Callbacks("once
memory"),k.Callbacks("once memory"),1,"rejected"]],i="pending",a={state:function(){return
i},always:function(){return s.done(arguments).fail(arguments),this},"catch":function(e){return
a.then(null,e)},pipe:function(){var i=arguments;return
k.Deferred(function(r){k.each(o,function(e,t){var n=m(i[t[4]])&&i[t[4]];s[t[1]](function(){var
e=n&&n.apply(this,arguments);e&&m(e.promise)?e.promise().progress(r.notify).done(r.resolve).fail(r.reject):r[t[0]+"With
u=0;function l(i,o,a,s){return function(){var n=this,r=arguments,e=function(){var
e,t;if(!(i<u)){if((e=a.apply(n,r))===o.promise())throw new TypeError("Thenable
self-resolution");t=e&&("object"==typeof e||"function"==typeof
e)&&e.then,m(t)?s?t.call(e,l(u,o,M,s),l(u,o,I,s)):(u++,t.call(e,l(u,o,M,s),l(u,o,I,s),l(u,o,M,o.notifyWith))):(a!==M&&
0,r=[e]),(s||o.resolveWith)(n,r))}},t=s?e:function(){try{e()}catch(e){k.Deferred.exceptionHook&&k.Deferred.exceptionHo
0,r=[e]),o.rejectWith(n,r))}};i?t():(k.Deferred.getStackHook&&(t.stackTrace=k.Deferred.getStackHook()),C.setTimeout(t)
k.Deferred(function(e){o[0][3].add(l(0,e,m(r)?r:M,e.notifyWith)),o[1][3].add(l(0,e,m(t)?t:M)),o[2][3].add(l(0,e,m(n)?n
null!=e?k.extend(e,a):a}},s={};return k.each(o,function(e,t){var
n=t[2],r=t[5];a[t[1]]=n.add,r&&n.add(function(){i=r},o[3-e][2].disable,o[3-e][3].disable,o[0][2].lock,o[0][3].lock),n.a
s[t[0]+"With"](this===s?void
0:this,arguments),this},s[t[0]+"With"]=n.fireWith}),a.promise(s),e&&e.call(s,s),s},when:function(e){var
n=arguments.length,t=n,r=Array(t),i=s.call(arguments),o=k.Deferred(),a=function(t){return
function(e){r[t]=this,i[t]=1<arguments.length?s.call(arguments):e,--n||o.resolveWith(r,i)}};if(n<=1&&(W(e,o.done(a(t))
o.then();while(t--)W(i[t],a(t),o.reject);return o.promise()}});var
$=/^(Eval|Internal|Range|Reference|Syntax|Type|URI)Error$/;k.Deferred.exceptionHook=function(e,t){C.console&&C.console
exception: "+e.message,e.stack,t)},k.readyException=function(e){C.setTimeout(function(){throw
e})};var F=k.Deferred();function
B(){E.removeEventListener("DOMContentLoaded",B),C.removeEventListener("load",B),k.ready()}k.fn.ready=function(e){return
```

```
F.then(e)["catch"](function(e){k.readyException(e)}),this},k.extend({isReady:!1,readyWait:1,ready:function(e){(!0===e?-
_=function(e,t,n,r,i,o,a){var s=0,u=e.length,l=null==n;if("object"===w(n))for(s in
i=!0,n)_(e,t,s,n[s],!0,o,a);else if(void
0!==r&&(i=!0,m(r)||(a=!0),l&&(a?(t.call(e,r),t=null):(l=t,t=function(e,t,n){return
l.call(k(e),n)})),t))for(;s<u;s++)t(e[s],n,a?r:r.call(e[s],s,t(e[s],n)));return
i?e:l?t.call(e):u?t(e[0],n):o},z=/^-ms-/,U=/-([a-z])/g;function X(e,t){return
t.toUpperCase()}function V(e){return e.replace(z,"ms-").replace(U,X)}var G=function(e){return
1===e.nodeType||9===e.nodeType||!+e.nodeType};function
Y(){this.expando=k.expando+Y.uid++}Y.uid=1,Y.prototype={cache:function(e){var
t=e[this.expando];return
t||(t={},G(e)&&(e.nodeType?e[this.expando]=t:Object.defineProperty(e,this.expando,{value:t,configurable:!0}))),t},set:
r,i=this.cache(e);if("string"==typeof t)i[V(t)]=n;else for(r in t)i[V(r)]=t[r];return
i},get:function(e,t){return void
0===t?this.cache(e):e[this.expando]&&e[this.expando][V(t)]},access:function(e,t,n){return void
0===t||t&&"string"==typeof t&&void 0===n?this.get(e,t):(this.set(e,t,n),void
0!==n?n:t)},remove:function(e,t){var n,r=e[this.expando];if(void 0!==r){if(void
0!==t){n=(t=Array.isArray(t)?t.map(V):(t=V(t))in r?[t]:t.match(R)||[]).length;while(n--)delete
r[t[n]]}(void 0===t||k.isEmptyObject(r))&&(e.nodeType?e[this.expando]=void 0:delete
e[this.expando])}},hasData:function(e){var t=e[this.expando];return void
0!==t&&!k.isEmptyObject(t)}};var Q=new Y,J=new
Y,K=/^(?:\{[\w\W]*\}|\[[\w\W]*\])$/,Z=/[A-Z]/g;function ee(e,t,n){var r,i;if(void
0===n&&1===e.nodeType)if(r="data-"+t.replace(Z,"-$&").toLowerCase(),"string"==typeof(n=e.getAttribute(r))){try{n="true"
n=void 0;return n}k.extend({hasData:function(e){return
J.hasData(e)||Q.hasData(e)},data:function(e,t,n){return
J.access(e,t,n)},removeData:function(e,t){J.remove(e,t)},_data:function(e,t,n){return
Q.access(e,t,n)},_removeData:function(e,t){Q.remove(e,t)}}),k.fn.extend({data:function(n,e){var
t,r,i,o=this[0],a=o&&o.attributes;if(void
0===n){if(this.length&&(i=J.get(o),1===o.nodeType&&!Q.get(o,"hasDataAttrs"))){t=a.length;while(t--)a[t]&&0===(r=a[t].na
i}return"object"==typeof n?this.each(function(){J.set(this,n)}):_(this,function(e){var t;if(o&&void
0===e)return void 0!==(t=J.get(o,n))?t:void 0!==(t=ee(o,n))?t:void
0;this.each(function(){J.set(this,n,e)})},null,e,1<arguments.length,null,!0)},removeData:function(e){return
this.each(function(){J.remove(this,e)})}}),k.extend({queue:function(e,t,n){var r;if(e)return
t=(t||"fx")+"queue",r=Q.get(e,t),n&&(!r||Array.isArray(n)?r=Q.access(e,t,k.makeArray(n)):r.push(n)),r||[]},dequeue:func
n=k.queue(e,t),r=n.length,i=n.shift(),o=k._queueHooks(e,t);"inprogress"===i&&(i=n.shift(),r--),i&&("fx"===t&&n.unshift
o.stop,i.call(e,function(){k.dequeue(e,t)},o)),!r&&o&&o.empty.fire()},_queueHooks:function(e,t){var
n=t+"queueHooks";return Q.get(e,n)||Q.access(e,n,{empty:k.Callbacks("once
memory").add(function(){Q.remove(e,[t+"queue",n])})})}}),k.fn.extend({queue:function(t,n){var
e=2;return"string"!=typeof t&&(n=t,t="fx",e--),arguments.length<e?k.queue(this[0],t):void
0===n?this:this.each(function(){var
e=k.queue(this,t,n);k._queueHooks(this,t),"fx"===t&&"inprogress"!==e[0]&&k.dequeue(this,t)})},dequeue:function(e){retu
this.each(function(){k.dequeue(this,e)})},clearQueue:function(e){return
this.queue(e||"fx",[])},promise:function(e,t){var
n,r=1,i=k.Deferred(),o=this,a=this.length,s=function(){--r||i.resolveWith(o,[o])};"string"!=typeof
e&&(t=e,e=void
0),e=e||"fx";while(a--)(n=Q.get(o[a],e+"queueHooks"))&&n.empty&&(r++,n.empty.add(s));return
s(),i.promise(t)}});var te=/[+-]?(?:\d*\.|)\d+(?:[eE][+-]?\d+|)/.source,ne=new
RegExp("^(?:([+-])=|)("+te+")([a-z%]*)$","i"),re=["Top","Right","Bottom","Left"],ie=E.documentElement,oe=function(e){re
k.contains(e.ownerDocument,e)},ae={composed:!0};ie.getRootNode&&(oe=function(e){return
k.contains(e.ownerDocument,e)||e.getRootNode(ae)===e.ownerDocument});var
se=function(e,t){return"none"===(e=t||e).style.display||""===e.style.display&&oe(e)&&"none"===k.css(e,"display")},ue=fu
i,o,a={};for(o in t)a[o]=e.style[o],e.style[o]=t[o];for(o in
i=n.apply(e,r||[]),t)e.style[o]=a[o];return i};function le(e,t,n,r){var
i,o,a=20,s=r?function(){return r.cur()}:function(){return
k.css(e,t,"")},u=s(),l=n&&n[3]||(k.cssNumber[t]?"":"px"),c=e.nodeType&&(k.cssNumber[t]||"px"!==l&&+u)&&ne.exec(k.css(e,
n&&(c=+c||+u||0,i=n[1]?c+(n[1]+1)*n[2]:+n[2],r&&(r.unit=l,r.start=c,r.end=i)),i}var ce={};function
fe(e,t){for(var
n,r,i,o,a,s,u,l=[],c=0,f=e.length;c<f;c++)(r=e[c]).style&&(n=r.style.display,t?("none"===n&&(l[c]=Q.get(r,"display")||n
0,a=(i=r).ownerDocument,s=i.nodeName,(u=ce[s])||(o=a.body.appendChild(a.createElement(s)),u=k.css(o,"display"),o.parent
e}k.fn.extend({show:function(){return fe(this,!0)},hide:function(){return
fe(this)},toggle:function(e){return"boolean"==typeof
e?e?this.show():this.hide():this.each(function(){se(this)?k(this).show():k(this).hide()})}});var
pe=/^(?:checkbox|radio)$/i,de=/<([a-z][^\/\0>\x20\t\r\n\f]*)/i,he=/^$|^module$|\/(?:java|ecma)script/i,ge={option:[1,"<
multiple='multiple'>","</select>"],thead:[1,"<table>","</table>"],col:[2,"<table><colgroup>","</colgroup></table>"],tr
ve(e,t){var n;return n="undefined"!=typeof
e.getElementsByTagName?e.getElementsByTagName(t||"*"):"undefined"!=typeof
e.querySelectorAll?e.querySelectorAll(t||"*"):[],void 0===t||t&&A(e,t)?k.merge([e],n):n}function
ye(e,t){for(var
n=0,r=e.length;n<r;n++)Q.set(e[n],"globalEval",!t||Q.get(t[n],"globalEval"))}ge.optgroup=ge.option,ge.tbody=ge.tfoot=ge
me,xe,be=/<|&#?\w+;/;function we(e,t,n,r,i){for(var
o,a,s,u,l,c,f=t.createDocumentFragment(),p=[],d=0,h=e.length;d<h;d++)if((o=e[d])||0===o)if("object"===w(o))k.merge(p,o
if(be.test(o)){a=a||f.appendChild(t.createElement("div")),s=(de.exec(o)||["",""])[1].toLowerCase(),u=ge[s]||ge._default
p.push(t.createTextNode(o));f.textContent="",d=0;while(o=p[d++])if(r&&-1<k.inArray(o,r))i&&i.push(o);else
if(l=oe(o),a=ve(f.appendChild(o),"script"),l&&ye(a),n){c=0;while(o=a[c++])he.test(o.type||"")&&n.push(o)}return
f}me=E.createDocumentFragment().appendChild(E.createElement("div")),(xe=E.createElement("input")).setAttribute("type","
Te=/^key/,Ce=/^(?:mouse|pointer|contextmenu|drag|drop)|click/,Ee=/^([^.]*)(?:\.(.+)|)/;function
ke(){return!0}function Se(){return!1}function Ne(e,t){return e===function(){try{return
E.activeElement}catch(e){}}()=="focus"===t}function Ae(e,t,n,r,i,o){var a,s;if("object"==typeof
t){for(s in"string"!=typeof n&&(r=r||n,n=void 0),t)Ae(e,s,n,r,t[s],o);return
e}if(null==r&&null==i?(i=n,r=n=void 0):null==i&&("string"==typeof n?(i=r,r=void 0):(i=r,r=n,n=void
0)),!1===i)i=Se;else if(!i)return e;return 1===o&&(a=i,(i=function(e){return
k().off(e),a.apply(this,arguments)}).guid=a.guid||(a.guid=k.guid++)),e.each(function(){k.event.add(this,t,i,r,n)})}func
De(e,i,o){o?(Q.set(e,i,!1),k.event.add(e,i,{namespace:!1,handler:function(e){var
t,n,r=Q.get(this,i);if(1&e.isTrigger&&this[i]){if(r.length)(k.event.special[i]||{}).delegateType&&e.stopPropagation();e
if(r=s.call(arguments),Q.set(this,i,r),t=o(this,i),this[i](),r!==(n=Q.get(this,i))||t?Q.set(this,i,!1):n={},r!==n)retu
e.stopImmediatePropagation(),e.preventDefault(),n.value}else
r.length&&(Q.set(this,i,{value:k.event.trigger(k.extend(r[0],k.Event.prototype),r.slice(1),this)}),e.stopImmediateProp
0===Q.get(e,i)&&k.event.add(e,i,ke)}k.event={global:{},add:function(t,e,n,r,i){var
```

```
o,a,s,u,l,c,f,p,d,h,g,v=Q.get(t);if(v){n.handler&&(n=(o=n).handler,i=o.selector),i&&k.find.matchesSelector(ie,i),n.guic
k&&k.event.triggered!==e.type?k.event.dispatch.apply(t,arguments):void
0}),l=(e=(e||"").match(R)||[""]).length;while(l--)d=g=(s=Ee.exec(e[l])||[])[1],h=(s[2]||"").split(".").sort(),d&&(f=k.e
o,a,s,u,l,c,f,p,d,h,g,v=Q.hasData(e)&&Q.get(e);if(v&&(u=v.events)){l=(t=(t||"").match(R)||[""]).length;while(l--)if(d=g
RegExp("(^|\\.)"+h.join("\\.(?:.*\\.|)")+"(\\.|$)"),a=o=p.length;while(o--)c=p[o],!i&&g!==c.origType||n&&n.guid!==c.gui
u[d])}else for(d in u)k.event.remove(e,d+t[l],n,r,!0);k.isEmptyObject(u)&&Q.remove(e,"handle
events")}},dispatch:function(e){var t,n,r,i,o,a,s=k.event.fix(e),u=new
Array(arguments.length),l=(Q.get(this,"events")||{})[s.type]||[],c=k.event.special[s.type]||{};for(u[0]=s,t=1;t<argumen
0!==(r=((k.event.special[o.origType]||{}).handle||o.handler).apply(i.elem,u))&&!1===(s.result=r)&&(s.preventDefault(),s
c.postDispatch&&c.postDispatch.call(this,s),s.result}},handlers:function(e,t){var
n,r,i,o,a,s=[],u=t.delegateCount,l=e.target;if(u&&l.nodeType&&!("click"===e.type&&1<=e.button))for(;l!==this;l=l.parent
0===a[i=(r=t[n]).selector+"
"]&&(a[i]=r.needsContext?-1<k(i,this).index(l):k.find(i,this,null,[l]).length),a[i]&&o.push(r);o.length&&s.push({elem:l
l=this,u<t.length&&s.push({elem:l,handlers:t.slice(u)}),s},addProp:function(t,e){Object.defineProperty(k.Event.prototyp
e(this.originalEvent)}:function(){if(this.originalEvent)return
this.originalEvent[t]},set:function(e){Object.defineProperty(this,t,{enumerable:!0,configurable:!0,writable:!0,value:e
e[k.expando]?e:new k.Event(e)},special:{load:{noBubble:!0},click:{setup:function(e){var
t=this||e;return pe.test(t.type)&&t.click&&A(t,"input")&&De(t,"click",ke),!1},trigger:function(e){var
t=this||e;return pe.test(t.type)&&t.click&&A(t,"input")&&De(t,"click"),!0},_default:function(e){var
t=e.target;return
pe.test(t.type)&&t.click&&A(t,"input")&&Q.get(t,"click")||A(t,"a")}},beforeunload:{postDispatch:function(e){void
0!==e.result&&e.originalEvent&&(e.originalEvent.returnValue=e.result)}}}},k.removeEvent=function(e,t,n){e.removeEventLi
instanceof k.Event))return new
k.Event(e,t);e&&e.type?(this.originalEvent=e,this.type=e.type,this.isDefaultPrevented=e.defaultPrevented||void
0===e.defaultPrevented&&!1===e.returnValue?ke:Se,this.target=e.target&&3===e.target.nodeType?e.target.parentNode:e.tarc
e=this.originalEvent;this.isDefaultPrevented=ke,e&&!this.isSimulated&&e.preventDefault()},stopPropagation:function(){va
e=this.originalEvent;this.isPropagationStopped=ke,e&&!this.isSimulated&&e.stopPropagation()},stopImmediatePropagation:
e=this.originalEvent;this.isImmediatePropagationStopped=ke,e&&!this.isSimulated&&e.stopImmediatePropagation(),this.stop
t=e.button;return null==e.which&&Te.test(e.type)?null!=e.charCode?e.charCode:e.keyCode:!e.which&&void
0!==t&&Ce.test(e.type)?1&t?1:2&t?3:4&t?2:0:e.which}},k.event.addProp),k.each({focus:"focusin",blur:"focusout"},function
De(this,e,Ne),!1},trigger:function(){return
De(this,e),!0},delegateType:t}}),k.each({mouseenter:"mouseover",mouseleave:"mouseout",pointerenter:"pointerover",pointe
t,n=e.relatedTarget,r=e.handleObj;return
n&&(n===this||k.contains(this,n))||(e.type=r.origType,t=r.handler.apply(this,arguments),e.type=i),t}}}),k.fn.extend({on
Ae(this,e,t,n,r)},one:function(e,t,n,r){return Ae(this,e,t,n,r,1)},off:function(e,t,n){var
r,i;if(e&&e.preventDefault&&e.handleObj)return
r=e.handleObj,k(e.delegateTarget).off(r.namespace?r.origType+"."+r.namespace:r.origType,r.selector,r.handler),this;if("
e){for(i in e)this.off(i,t,e[i]);return this}return!1!==t&&"function"!=typeof t||(n=t,t=void
0),!1===n&&(n=Se),this.each(function(){k.event.remove(this,e,n,t)})}}});var
je=/<(?!area|br|col|embed|hr|img|input|link|meta|param)(([a-z][^\/\0>\x20\t\r\n\f]*)[^>]*)\/>/gi,qe=/<script|<style|<li
Oe(e,t){return
A(e,"table")&&A(11!==t.nodeType?t:t.firstChild,"tr")&&k(e).children("tbody")[0]||e}function
Pe(e){return e.type=(null!==e.getAttribute("type"))+"/"+e.type,e}function
Re(e){return"true/"===(e.type||"").slice(0,5)?e.type=e.type.slice(5):e.removeAttribute("type"),e}function
Me(e,t){var
n,r,i,o,a,s,u,l;if(1===t.nodeType){if(Q.hasData(e)&&(o=Q.access(e),a=Q.set(t,o),l=o.events))for(i in
delete
a.handle,a.events={},l)for(n=0,r=l[i].length;n<r;n++)k.event.add(t,i,l[i][n]);J.hasData(e)&&(s=J.access(e),u=k.extend(
Ie(n,r,i,o){r=g.apply([],r);var
e,t,a,s,u,l,c=0,f=n.length,p=f-1,d=r[0],h=m(d);if(h||1<f&&"string"==typeof
d&&!y.checkClone&&Le.test(d))return n.each(function(e){var
t=n.eq(e);h&&(r[0]=d.call(this,e,t.html())),Ie(t,r,i,o)});if(f&&(t=(e=we(r,n[0].ownerDocument,!1,n,o)).firstChild,1===e
n}function We(e,t,n){for(var
r,i=t?k.filter(t,e):e,o=0;null!=(r=i[o]);o++)n||1!==r.nodeType||k.cleanData(ve(r)),r.parentNode&&(n&&oe(r)&&ye(ve(r,"sc
e}k.extend({htmlPrefilter:function(e){return e.replace(je,"<$1></$2>")},clone:function(e,t,n){var
r,i,o,a,s,u,l,c=e.cloneNode(!0),f=oe(e);if(!(y.noCloneChecked||1!==e.nodeType&&11!==e.nodeType||k.isXMLDoc(e)))for(a=ve
0,"input"===(l=u.nodeName.toLowerCase())&&pe.test(s.type)?u.checked=s.checked:"input"!==l&&"textarea"!==l||(u.defaultVa
Me(e,c);return 0<(a=ve(c,"script")).length&&ye(a,!f&&ve(e,"script")),c},cleanData:function(e){for(var
t,n,r,i=k.event.special,o=0;void 0!==(n=e[o]);o++)if(G(n)){if(t=n[Q.expando]){if(t.events)for(r in
t.events)i[r]?k.event.remove(n,r):k.removeEvent(n,r,t.handle);n[Q.expando]=void
0}n[J.expando]&&(n[J.expando]=void 0)}}}),k.fn.extend({detach:function(e){return
We(this,e,!0)},remove:function(e){return We(this,e)},text:function(e){return
_(this,function(e){return void
0===e?k.text(this):this.empty().each(function(){1!==this.nodeType&&11!==this.nodeType&&9!==this.nodeType||(this.textCon
Ie(this,arguments,function(e){1!==this.nodeType&&11!==this.nodeType&&9!==this.nodeType||Oe(this,e).appendChild(e)})},pr
Ie(this,arguments,function(e){if(1===this.nodeType||11===this.nodeType||9===this.nodeType){var
t=Oe(this,e);t.insertBefore(e,t.firstChild)}})},before:function(){return
Ie(this,arguments,function(e){this.parentNode&&this.parentNode.insertBefore(e,this)})},after:function(){return
Ie(this,arguments,function(e){this.parentNode&&this.parentNode.insertBefore(e,this.nextSibling)})},empty:function(){for
e,t=0;null!=(e=this[t]);t++)1===e.nodeType&&(k.cleanData(ve(e,!1)),e.textContent="");return
this},clone:function(e,t){return e=null!=e&&e,t=null==t?e:t,this.map(function(){return
k.clone(this,e,t)})},html:function(e){return _(this,function(e){var
t=this[0]||{},n=0,r=this.length;if(void 0===e&&1===t.nodeType)return t.innerHTML;if("string"==typeof
e&&!qe.test(e)&&!ge[(de.exec(e)||["",""])[1].toLowerCase()]){e=k.htmlPrefilter(e);try{for(;n<r;n++)1===(t=this[n]||{})
n=[];return Ie(this,arguments,function(e){var
t=this.parentNode;k.inArray(this,n)<0&&(k.cleanData(ve(this)),t&&t.replaceChild(e,this))},n)}}),k.each({appendTo:"appen
t,n=[],r=k(e),i=r.length-1,o=0;o<=i;o++)t=o===i?this:this.clone(!0),k(r[o])[a](t),u.apply(n,t.get());return
this.pushStack(n)}});var $e=new RegExp("^("+te+")(?!px)[a-z%]+$","i"),Fe=function(e){var
t=e.ownerDocument.defaultView;return t&&t.opener||(t=C),t.getComputedStyle(e)},Be=new
RegExp(re.join("|"),"i");function _e(e,t,n){var
r,i,o,a,s=e.style;return(n=n||Fe(e))&&(""!==(a=n.getPropertyValue(t)||n[t])||oe(e)||(a=k.style(e,t)),!y.pixelBoxStyles
0!==a?a+"":a}function
ze(e,t){return{get:function(){if(!e())return(this.get=t).apply(this,arguments);delete
this.get}}}!function(){function
e(){if(u){s.style.cssText="position:absolute;left:-11111px;width:60px;margin-top:1px;padding:0;border:0",u.style.cssTex
e=C.getComputedStyle(u);n="1%"!==e.top,a=12===t(e.marginLeft),u.style.right="60%",o=36===t(e.right),r=36===t(e.width),u
t(e){return Math.round(parseFloat(e))}var
```

```
n,r,i,o,a,s=E.createElement("div"),u=E.createElement("div");u.style&&(u.style.backgroundClip="content-box",u.cloneNode
e(),r},pixelBoxStyles:function(){return e(),o},pixelPosition:function(){return
e(),n},reliableMarginLeft:function(){return e(),a},scrollboxSize:function(){return e(),i}}))}();var
Ue=["Webkit","Moz","ms"],Xe=E.createElement("div").style,Ve={};function Ge(e){var
t=k.cssProps[e]||Ve[e];return t||(e in Xe?e:Ve[e]=function(e){var
t=e[0].toUpperCase()+e.slice(1),n=Ue.length;while(n--)if((e=Ue[n]+t)in Xe)return e}(e)||e)}var
Ye=/^(none|table(?!-c[ea]).+)/,Qe=/^--/,Je={position:"absolute",visibility:"hidden",display:"block"},Ke={letterSpacing
Ze(e,t,n){var r=ne.exec(t);return r?Math.max(0,r[2]-(n||0))+(r[3]||"px"):t}function
et(e,t,n,r,i,o){var a="width"===t?1:0,s=0,u=0;if(n===(r?"border":"content"))return
0;for(;a<4;a+=2)"margin"===n&&(u+=k.css(e,n+re[a],!0,i)),r?("content"===n&&(u-=k.css(e,"padding"+re[a],!0,i)),"margin"
tt(e,t,n){var
r=Fe(e),i=(!y.boxSizingReliable()||n)&&"border-box"===k.css(e,"boxSizing",!1,r),o=i,a=_e(e,t,r),s="offset"+t[0].toUppe
a;a="auto"}return(!y.boxSizingReliable()&&i||"auto"===a||!parseFloat(a)&&"inline"===k.css(e,"display",!1,r))&&e.getClie
in e)&&(a=e[s])),(a=parseFloat(a)||0)+et(e,t,n||(i?"border":"content"),o,r,a)+"px"}function
nt(e,t,n,r,i){return new
nt.prototype.init(e,t,n,r,i)}k.extend({cssHooks:{opacity:{get:function(e,t){if(t){var
n=_e(e,"opacity");return""===n?"1":n}}}},cssNumber:{animationIterationCount:!0,columnCount:!0,fillOpacity:!0,flexGrow:
i,o,a,s=V(t),u=Qe.test(t),l=e.style;if(u||(t=Ge(s)),a=k.cssHooks[t]||k.cssHooks[s],void 0===n)return
a&&"get"in a&&void 0!==(i=a.get(e,!1,r))?i:l[t];"string"===(o=typeof
n)&&(i=ne.exec(n))&&i[1]&&(n=le(e,t,i),o="number"),null!=n&&n==n&&("number"!==o||u||(n+=i&&i[3]||(k.cssNumber[s]?"":"p
a&&void 0===(n=a.set(e,n,r))||(u?l.setProperty(t,n):l[t]=n))}},css:function(e,t,n,r){var
i,o,a,s=V(t);return Qe.test(t)||(t=Ge(s)),(a=k.cssHooks[t]||k.cssHooks[s])&&"get"in
a&&(i=a.get(e,!0,n)),void 0===i&&(i=_e(e,t,r)),"normal"===i&&t in
Ke&&(i=Ke[t]),""===n||n?(o=parseFloat(i),!0===n||isFinite(o)?o||0:i):i}}),k.each(["height","width"],function(e,u){k.cs
tt(e,u,n)})},set:function(e,t,n){var
r,i=Fe(e),o=!y.scrollboxSize()&&"absolute"===i.position,a=(o||n)&&"border-box"===k.css(e,"boxSizing",!1,i),s=n?et(e,u,
a&&o&&(s-=Math.ceil(e["offset"+u[0].toUpperCase()+u.slice(1)]-parseFloat(i[u])-et(e,u,"border",!1,i)-.5)),s&&(r=ne.exec
e.getBoundingClientRect().left}))+"px"}),k.each({margin:"",padding:"",border:"Width"},function(i,o){k.cssHooks[i+o]={e
t=0,n={},r="string"==typeof e?e.split(" "):[e];t<4;t++)n[i+re[t]+o]=r[t]||r[t-2]||r[0];return
n}},"margin"!==i&&(k.cssHooks[i+o].set=Ze)},k.fn.extend({css:function(e,t){return
_(this,function(e,t,n){var
r,i,o={},a=0;if(Array.isArray(t)){for(r=Fe(e),i=t.length;a<i;a++)o[t[a]]=k.css(e,t[a],!1,r);return
o}return void
0!==n?k.style(e,t,n):k.css(e,t)},e,t,1<arguments.length)}}),((k.Tween=nt).prototype={constructor:nt,init:function(e,t,
e=nt.propHooks[this.prop];return
e&&e.get?e.get(this):nt.propHooks._default.get(this)},run:function(e){var
t,n=nt.propHooks[this.prop];return
this.options.duration?this.pos=t=k.easing[this.easing](e,this.options.duration*e,0,1,this.options.duration):this.pos=t=
t;return
1!==e.elem.nodeType||null!=e.elem[e.prop]&&null==e.elem.style[e.prop]?e.elem[e.prop]:(t=k.css(e.elem,e.prop,""))&&"auto
e},swing:function(e){return.5-Math.cos(e*Math.PI)/2},_default:"swing"},k.fx=nt.prototype.init,k.fx.step={};var
rt,it,ot,at,st=/^(?:toggle|show|hide)$/,ut=/queueHooks$/;function
lt(){it&&(!1===E.hidden&&C.requestAnimationFrame?C.requestAnimationFrame(lt):C.setTimeout(lt,k.fx.interval),k.fx.tick()
ct(){return C.setTimeout(function(){rt=void 0}),rt=Date.now()}function ft(e,t){var
n,r=0,i={height:e};for(t=t?1:0;r<4;r+=2-t)i["margin"+(n=re[r])]=i["padding"+n]=e;return
t&&(i.opacity=i.width=e),i}function pt(e,t,n){for(var
r,i=(dt.tweeners[t]||[]).concat(dt.tweeners["*"]),o=0,a=i.length;o<a;o++)if(r=i[o].call(n,t,e))return
r}function dt(o,e,t){var n,a,r=0,i=dt.prefilters.length,s=k.Deferred().always(function(){delete
u.elem},u=function(){if(a)return!1;for(var
e=rt||ct(),t=Math.max(0,l.startTime+l.duration-e),n=1-(t/l.duration||0),r=0,i=l.tweens.length;r<i;r++)l.tweens[r].run(
s.notifyWith(o,[l,n,t]),n<1&&i?t:(i||s.notifyWith(o,[l,1,0]),s.resolveWith(o,[l]),!1)},l=s.promise({elem:o,props:k.exte
n=k.Tween(o,l.opts,e,t,l.opts.specialEasing[e]||l.opts.easing);return
l.tweens.push(n),n},stop:function(e){var t=0,n=e?l.tweens.length:0;if(a)return
this;for(a=!0;t<n;t++)l.tweens[t].run(1);return
e?(s.notifyWith(o,[l,1,0]),s.resolveWith(o,[l,e])):s.rejectWith(o,[l,e]),this}}),c=l.props;for(!function(e,t){var
n,r,i,o,a;for(n in
e)if(i=t[r=V(n)],o=e[n],Array.isArray(o)&&(i=o[1],o=e[n]=o[0]),n!==r&&(e[r]=o,delete
e[n]),(a=k.cssHooks[r])&&"expand"in a)for(n in o=a.expand(o),delete e[r],o)n in
e||(e[n]=o[n],t[n]=i);else
t[r]=i}(c,l.opts.specialEasing);r<i;r++)if(n=dt.prefilters[r].call(l,o,c,l.opts))return
m(n.stop)&&(k._queueHooks(l.elem,l.opts.queue).stop=n.stop.bind(n)),n;return
k.map(c,pt,l),m(l.opts.start)&&l.opts.start.call(o,l),l.progress(l.opts.progress).done(l.opts.done,l.opts.complete).fa
n=this.createTween(e,t);return
le(n.elem,e,ne.exec(t),n),n}]},tweener:function(e,t){m(e)?(t=e,e=["*"]):e=e.match(R);for(var
n,r=0,i=e.length;r<i;r++)n=e[r],dt.tweeners[n]=dt.tweeners[n]||[],dt.tweeners[n].unshift(t)},prefilters:[function(e,t,
r,i,o,a,s,u,l,c,f="width"in t||"height"in
t,p=this,d={},h=e.style,g=e.nodeType&&se(e),v=Q.get(e,"fxshow");for(r in
n.queue||(null==(a=k._queueHooks(e,"fx")).unqueued&&(a.unqueued=0,s=a.empty.fire,a.empty.fire=function(){a.unqueued||s
t[r],o=o||"toggle"===i,i===(g?"hide":"show")){if("show"!==i||!v||void
0===v[r])continue;g=!0}d[r]=v&&v[r]||k.style(e,r)}if((u=!k.isEmptyObject(t))||!k.isEmptyObject(d))for(r
in
f&&1===e.nodeType&&(n.overflow=[h.overflow,h.overflowX,h.overflowY],null==(l=v&&v.display)&&(l=Q.get(e,"display")),"no
v&&(g=v.hidden):v=Q.access(e,"fxshow",{display:l}),o&&(v.hidden=!g),g&&fe([e],!0),p.done(function(){for(r
in g||fe([e]),Q.remove(e,"fxshow"),d)k.style(e,r,d[r])})),u=pt(g?v[r]:0,r,p),r in
v||(v[r]=u.start,g&&(u.end=u.start,u.start=0))}],prefilter:function(e,t){t?dt.prefilters.unshift(e):dt.prefilters.push
r=e&&"object"==typeof
e?k.extend({},e):{complete:n||!n&&t||m(e)&&e,duration:e,easing:n&&t||t&&!m(t)&&t};return
k.fx.off?r.duration=0:"number"!=typeof r.duration&&(r.duration in
k.fx.speeds?r.duration=k.fx.speeds[r.duration]:r.duration=k.fx.speeds._default),null!=r.queue&&!0!==r.queue||(r.queue=
this.filter(se).css("opacity",0).show().end().animate({opacity:t},e,n,r)},animate:function(t,e,n,r){var
i=k.isEmptyObject(t),o=k.speed(e,n,r),a=function(){var
e=dt(this,k.extend({},t),o);(i||Q.get(this,"finish"))&&e.stop(!0)};return
a.finish=a,i||!1===o.queue?this.each(a):this.queue(o.queue,a)},stop:function(i,e,o){var
a=function(e){var t=e.stop;delete e.stop,t(o)};return"string"!=typeof i&&(o=e,e=i,i=void
0),e&&!1!==i&&this.queue(i||"fx",[]),this.each(function(){var
e=!0,t=null!=i&&i+"queueHooks",n=k.timers,r=Q.get(this);if(t)r[t]&&r[t].stop&&a(r[t]);else for(t in
r)r[t]&&r[t].stop&&ut.test(t)&&a(r[t]);for(t=n.length;t--;)n[t].elem!==this||null!=i&&n[t].queue!==i||(n[t].anim.stop(
```

```
e,t=Q.get(this),n=t[a+"queue"],r=t[a+"queueHooks"],i=k.timers,o=n?n.length:0;for(t.finish=!0,k.queue(this,a,[]),r&&r.st
t.finish})}}),k.each(["toggle","show","hide"],function(e,r){var
i=k.fn[r];k.fn[r]=function(e,t,n){return null==e||"boolean"==typeof
e?i.apply(this,arguments):this.animate(ft(r,!0),e,t,n)}}),k.each({slideDown:ft("show"),slideUp:ft("hide"),slideToggle:
this.animate(r,e,t,n)}}),k.timers=[],k.fx.tick=function(){var
e,t=0,n=k.timers;for(rt=Date.now();t<n.length;t++)(e=n[t])()||n[t]!==e||n.splice(t--,1);n.length||k.fx.stop(),rt=void
0},k.fx.timer=function(e){k.timers.push(e),k.fx.start()},k.fx.interval=13,k.fx.start=function(){it||(it=!0,lt())},k.fx
r=k.fx&&k.fx.speeds[r]||r,e=e||"fx",this.queue(e,function(e,t){var
n=C.setTimeout(e,r);t.stop=function(){C.clearTimeout(n)}}})},ot=E.createElement("input"),at=E.createElement("select").ap
ht,gt=k.expr.attrHandle;k.fn.extend({attr:function(e,t){return
_(this,k.attr,e,t,1<arguments.length)},removeAttr:function(e){return
this.each(function(){k.removeAttr(this,e)})}}),k.extend({attr:function(e,t,n){var
r,i,o=e.nodeType;if(3!==o&&8!==o&&2!==o)return"undefined"==typeof
e.getAttribute?k.prop(e,t,n):(1===o&&k.isXMLDoc(e)||(i=k.attrHooks[t.toLowerCase()]||(k.expr.match.bool.test(t)?ht:void
0)),void 0!==n?null===n?void k.removeAttr(e,t):i&&"set"in i&&void
0!==(r=i.set(e,n,t))?r:(e.setAttribute(t,n+""),n):i&&"get"in
i&&null!==(r=i.get(e,t))?r:null==(r=k.find.attr(e,t))?void
0:r)},attrHooks:{type:{set:function(e,t){if(!y.radioValue&&"radio"===t&&A(e,"input")){var
n=e.value;return e.setAttribute("type",t),n&&(e.value=n),t}}}},removeAttr:function(e,t){var
n,r=0,i=t&&t.match(R);if(i&&1===e.nodeType)while(n=i[r++])e.removeAttribute(n)}}),ht={set:function(e,t,n){return!1===t
a=gt[t]||k.find.attr;gt[t]=function(e,t,n){var r,i,o=t.toLowerCase();return
n||(i=gt[o],gt[o]=r,r=null!=a(e,t,n)?o:null,gt[o]=i),r}});var
vt=/^(?:input|select|textarea|button)$/i,yt=/^(?:a|area)$/i;function
mt(e){return(e.match(R)||[]).join(" ")}function xt(e){return
e.getAttribute&&e.getAttribute("class")||""}function bt(e){return Array.isArray(e)?e:"string"==typeof
e&&e.match(R)||[]}k.fn.extend({prop:function(e,t){return
_(this,k.prop,e,t,1<arguments.length)},removeProp:function(e){return this.each(function(){delete
this[k.propFix[e]||e]})}}),k.extend({prop:function(e,t,n){var
r,i,o=e.nodeType;if(3!==o&&8!==o&&2!==o)return
1===o&&k.isXMLDoc(e)||(t=k.propFix[t]||t,i=k.propHooks[t]),void 0!==n?i&&"set"in i&&void
0!==(r=i.set(e,n,t))?r:e[t]=n:i&&"get"in
i&&null!==(r=i.get(e,t))?r:e[t]},propHooks:{tabIndex:{get:function(e){var
t=k.find.attr(e,"tabindex");return
t?parseInt(t,10):vt.test(e.nodeName)||yt.test(e.nodeName)&&e.href?0:-1}}},propFix:{"for":"htmlFor","class":"className"
t=e.parentNode;return t&&t.parentNode&&t.parentNode.selectedIndex,null},set:function(e){var
t=e.parentNode;t&&(t.selectedIndex,t.parentNode&&t.parentNode.selectedIndex)}}),k.each(["tabIndex","readOnly","maxLengt
e,n,r,i,o,a,s,u=0;if(m(t))return
this.each(function(e){k(this).addClass(t.call(this,e,xt(this)))});if((e=bt(t)).length)while(n=this[u++])if(i=xt(n),r=1=
"+mt(i)+" "){a=0;while(o=e[a++])r.indexOf(" "+o+" ")<0&&(r+=o+"
");i!==(s=mt(r))&&n.setAttribute("class",s)}return this},removeClass:function(t){var
e,n,r,i,o,a,s,u=0;if(m(t))return
this.each(function(e){k(this).removeClass(t.call(this,e,xt(this)))});if(!arguments.length)return
this.attr("class","");if((e=bt(t)).length)while(n=this[u++])if(i=xt(n),r=1===n.nodeType&&" "+mt(i)+"
"){a=0;while(o=e[a++])while(-1<r.indexOf(" "+o+" "))r=r.replace(" "+o+" ","
");i!==(s=mt(r))&&n.setAttribute("class",s)}return this},toggleClass:function(i,t){var o=typeof
i,a="string"===o||Array.isArray(i);return"boolean"==typeof
t&&a?t?this.addClass(i):this.removeClass(i):m(i)?this.each(function(e){k(this).toggleClass(i.call(this,e,xt(this),t),t)
e,t,n,r;if(a){t=0,n=k(this),r=bt(i);while(e=r[t++])n.hasClass(e)?n.removeClass(e):n.addClass(e)}else
void
0!==i&&"boolean"!==o||((e=xt(this))&&Q.set(this,"__className__",e),this.setAttribute&&this.setAttribute("class",e||!1=
t,n,r=0;t=" "+e+" ";while(n=this[r++])if(1===n.nodeType&&-1<(" "+mt(xt(n))+"
").indexOf(t))return!0;return!1}});var wt=/\r/g;k.fn.extend({val:function(n){var
r,e,i,t=this[0];return arguments.length?(i=m(n),this.each(function(e){var
t;1===this.nodeType&&(null==(t=i?n.call(this,e,k(this).val()):n)?t="":"number"==typeof
t?t+="":Array.isArray(t)&&(t=k.map(t,function(e){return
null==e?"":e+""})),(r=k.valHooks[this.type]||k.valHooks[this.nodeName.toLowerCase()])&&"set"in
r&&void
0!==r.set(this,t,"value")||(this.value=t))})):t?(r=k.valHooks[t.type]||k.valHooks[t.nodeName.toLowerCase()])&&"get"in
r&&void 0!==(e=r.get(t,"value"))?e:"string"==typeof(e=t.value)?e.replace(wt,""):null==e?"":e:void
0}}),k.extend({valHooks:{option:{get:function(e){var t=k.find.attr(e,"value");return
null!=t?t:mt(k.text(e))}},select:{get:function(e){var
t,n,r,i=e.options,o=e.selectedIndex,a="select-one"===e.type,s=a?null:[],u=a?o+1:i.length;for(r=o<0?u:a?o:0;r<u;r++)if(
t;s.push(t)}return s},set:function(e,t){var
n,r,i=e.options,o=k.makeArray(t),a=i.length;while(a--)((r=i[a]).selected=-1<k.inArray(k.valHooks.option.get(r),o))&&(n=
n||(e.selectedIndex=-1),o}}}}),k.each(["radio","checkbox"],function(){k.valHooks[this]={set:function(e,t){if(Array.isA
e.checked=-1<k.inArray(k(e).val(),t)}},y.checkOn||(k.valHooks[this].get=function(e){return
null===e.getAttribute("value")?"on":e.value})}),y.focusin="onfocusin"in C;var
Tt=/^(?:focusinfocus|focusoutblur)$/,Ct=function(e){e.stopPropagation()};k.extend(k.event,{trigger:function(e,t,n,r){va
i,o,a,s,u,l,c,f,p=[n||E],d=v.call(e,"type")?e.type:e,h=v.call(e,"namespace")?e.namespace.split(".")=[];if(o=f=a=n=n||E,
k.Event(d,"object"==typeof
e&&e)).isTrigger=r?2:3,e.namespace=h.join(".),e.rnamespace=e.namespace?new
RegExp("(^|\\.)"+h.join("\\.(?:.*\\.|)")+"(\\.|$)"):null,e.result=void
0,e.target||(e.target=n),t=null==t?[e]:k.makeArray(t,[e]),c=k.event.special[d]||{},r||!c.trigger||!1!==c.trigger.apply
e.type=d,r||e.isDefaultPrevented()||c._default&&!1!==c._default.apply(p.pop(),t)||!G(n)||u&&m(n[d])&&!x(n)&&((a=n[u])&
0,a&&(n[u]=a)),e.result}},simulate:function(e,t,n){var r=k.extend(new
k.Event,n,{type:e,isSimulated:!0});k.event.trigger(r,null,t)}}),k.fn.extend({trigger:function(e,t){return
this.each(function(){k.event.trigger(e,t,this)})},triggerHandler:function(e,t){var
n=this[0];if(n)return
k.event.trigger(e,t,n,!0)}}),y.focusin||k.each({focus:"focusin",blur:"focusout"},function(n,r){var
i=function(e){k.event.simulate(r,e.target,k.event.fix(e))};k.event.special[r]={setup:function(){var
e=this.ownerDocument||this,t=Q.access(e,r);t||e.addEventListener(n,i,!0),Q.access(e,r,(t||0)+1)},teardown:function(){va
e=this.ownerDocument||this,t=Q.access(e,r)-1;t?Q.access(e,r,t):(e.removeEventListener(n,i,!0),Q.remove(e,r))}}});var
Et=C.location,kt=Date.now(),St=/\?/;k.parseXML=function(e){var t;if(!e||"string"!=typeof e)return
null;try{t=(new C.DOMParser).parseFromString(e,"text/xml")}catch(e){t=void 0}return
t&&!t.getElementsByTagName("parsererror").length||k.error("Invalid XML: "+e),t};var
Nt=/\[\]$/,At=/\r?\n/g,Dt=/^(?:submit|button|image|reset|file)$/i,jt=/^(?:input|select|textarea|keygen)/i;function
qt(n,e,r,i){var
```

```
t;if(Array.isArray(e))k.each(e,function(e,t){r||Nt.test(n)?i(n,t):qt(n+"["+("object"==typeof
t&&null!=t?e:"")+"]",t,r,i)});else if(r||"object"!==w(e))i(n,e);else for(t in
e)qt(n+"["+t+"]",e[t],r,i)}k.param=function(e,t){var n,r=[],i=function(e,t){var
n=m(t)?t():t;r[r.length]=encodeURIComponent(e)+"="+encodeURIComponent(null==n?"":n)};if(null==e)return"";if(Array.isArr
for(n in e)qt(n,e[n],t,i);return r.join("&")},k.fn.extend({serialize:function(){return
k.param(this.serializeArray())},serializeArray:function(){return this.map(function(){var
e=k.prop(this,"elements");return e?k.makeArray(e):this}).filter(function(){var e=this.type;return
this.name&&!k(this).is(":disabled")&&jt.test(this.nodeName)&&!Dt.test(e)&&(this.checked||!pe.test(e))}).map(function(e,
n=k(this).val();return
null==n?null:Array.isArray(n)?k.map(n,function(e){return{name:t.name,value:e.replace(At,"\r\n")}}):{name:t.name,value:n
Lt=/%20/g,Ht=/#.*$/,Ot=/([?&])_=[^&]*/,Pt=/^(.*?):[
\t]*([^\r\n]*)$/gm,Rt=/^(?:GET|HEAD)$/,Mt=/^\/\//,It={},Wt={},$t="*/".concat("*"),Ft=E.createElement("a");function
Bt(o){return function(e,t){"string"!=typeof e&&(t=e,e="*");var
n,r=0,i=e.toLowerCase().match(R)||[];if(m(t))while(n=i[r++])"+"===n[0]?(n=n.slice(1)||"*",(o[n]=o[n]||[]).unshift(t)):
_t(t,i,o,a){var s={},u=t===Wt;function l(e){var r;return s[e]=!0,k.each(t[e]||[],function(e,t){var
n=t(i,o,a);return"string"!=typeof n||u||s[n]?u?!(r=n):void
0:(i.dataTypes.unshift(n),l(n),!1)}),r}return l(i.dataTypes[0])||!s["*"]&&l("*")}function zt(e,t){var
n,r,i=k.ajaxSettings.flatOptions||{};for(n in t)void 0!==t[n]&&((i[n]?e:r||(r={}))[n]=t[n]);return
r&&k.extend(!0,e,r),e}Ft.href=Et.href,k.extend({active:0,lastModified:{},etag:{},ajaxSettings:{url:Et.href,type:"GET",i
charset=UTF-8",accepts:{"*":$t,text:"text/plain",html:"text/html",xml:"application/xml,
text/xml",json:"application/json,
text/javascript"},contents:{xml:/\bxml\b/,html:/\bhtml/,json:/\bjson\b/},responseFields:{xml:"responseXML",text:"respon
text":String,"text html":!0,"text json":JSON.parse,"text
xml":k.parseXML},flatOptions:{url:!0,context:!0}},ajaxSetup:function(e,t){return
t?zt(zt(e,k.ajaxSettings),t):zt(k.ajaxSettings,e)},ajaxPrefilter:Bt(It),ajaxTransport:Bt(Wt),ajax:function(e,t){"object
e&&(t=e,e=void 0),t=t||{};var
c,f,p,n,d,r,h,g,i,o,v=k.ajaxSetup({},t),y=v.context||v,m=v.context&&(y.nodeType||y.jquery)?k(y):k.event,x=k.Deferred(),
memory"),w=v.statusCode||{},a={},s={},u="canceled",T={readyState:0,getResponseHeader:function(e){var
t;if(h){if(!n){n={};while(t=Pt.exec(p))n[t[1].toLowerCase()+" "]=(n[t[1].toLowerCase()+"
"]||[]).concat(t[2])}t=n[e.toLowerCase()+" "]}return null==t?null:t.join(",
")},getAllResponseHeaders:function(){return h?p:null},setRequestHeader:function(e,t){return
null==h&&(e=s[e.toLowerCase()]=s[e.toLowerCase()]||e,a[e]=t),this},overrideMimeType:function(e){return
null==h&&(v.mimeType=e),this},statusCode:function(e){var t;if(e)if(h)T.always(e[T.status]);else for(t
in e)w[t]=[w[t],e[t]];return this},abort:function(e){var t=e||u;return
c&&c.abort(t),l(0,t),this}};if(x.promise(T),v.url=((e||v.url||Et.href)+"").replace(Mt,Et.protocol+"//"),v.type=t.metho
v.data&&(v.data=k.param(v.data,v.traditional)),_t(It,v,t,T),h)return T;for(i
in(g=k.event&&v.global)&&0==k.active++&&k.event.trigger("ajaxStart"),v.type=v.type.toUpperCase(),v.hasContent=!Rt.test
v.data)&&(f+=(St.test(f)?"&":"?")+v.data,delete
v.data),!1===v.cache&&(f=f.replace(Ot,"$1"),o=(St.test(f)?"&":"?")+"_="+kt+++o),v.url=f+o),v.ifModified&&(k.lastModifie
"+$t+";
q=0.01":""):v.accepts["*"]),v.headers)T.setRequestHeader(i,v.headers[i]);if(v.beforeSend&&(!1===v.beforeSend.call(y,T,v
T.abort();if(u="abort",b.add(v.complete),T.done(v.success),T.fail(v.error),c=_t(Wt,v,t,T)){if(T.readyState=1,g&&m.trigg
T;v.async&&0<v.timeout&&(d=C.setTimeout(function(){T.abort("timeout")},v.timeout));try{h=!1,c.send(a,l)}catch(e){if(h)t
e;l(-1,e)}}else l(-1,"No Transport");function l(e,t,n,r){var
i,o,a,s,u,l=t;h||(h=!0,d&&C.clearTimeout(d),c=void
0,p=r||"",T.readyState=0<e?4:0,i=200<=e&&e<300||304===e,n&&(s=function(e,t,n){var
r,i,o,a,s=e.contents,u=e.dataTypes;while("*"===u[0])u.shift(),void
0===r&&(r=e.mimeType||t.getResponseHeader("Content-Type"));if(r)for(i in
s)if(s[i]&&s[i].test(r)){u.unshift(i);break}if(u[0]in n)o=u[0];else{for(i in
n){if(!u[0]||e.converters[i+" "+u[0]]){o=i;break}a||(a=i)}o=o||a}if(o)return
o!==u[0]&&u.unshift(o),n[o]}(v,T,n)),s=function(e,t,n,r){var
i,o,a,s,u,l={},c=e.dataTypes.slice();if(c[1])for(a in
e.converters)l[a.toLowerCase()]=e.converters[a];o=c.shift();while(o)if(e.responseFields[o]&&(n[e.responseFields[o]]=t),
if("*"!==u&&u!==o){if(!(a=l[u+" "+o]||l["* "+o]))for(i in l)if((s=i.split(" "))[1]===o&&(a=l[u+"
"+s[0]]||l["*
"+s[0]])){!0===a?a=l[i]:!0!==l[i]&&(o=s[0],c.unshift(s[1]));break}if(!0!==a)if(a&&e["throws"])t=a(t);else
try{t=a(t)}catch(e){return{state:"parsererror",error:a?e:"No conversion from "+u+" to
"+o}}}return{state:"success",data:t}}(v,s,T,i),i?(v.ifModified&&((u=T.getResponseHeader("Last-Modified"))&&(k.lastModi
0,g&&m.trigger(i?"ajaxSuccess":"ajaxError",[T,v,i?o:a]),b.fireWith(y,[T,l]),g&&(m.trigger("ajaxComplete",[T,v]),--k.act
T},getJSON:function(e,t,n){return k.get(e,t,n,"json")},getScript:function(e,t){return k.get(e,void
0,t,"script")}}),k.each(["get","post"],function(e,i){k[i]=function(e,t,n,r){return
m(t)&&(r=r||n,n=t,t=void
0),k.ajax(k.extend({url:e,type:i,dataType:r,data:t,success:n},k.isPlainObject(e)&&e))}}),k._evalUrl=function(e,t){retur
k.ajax({url:e,type:"GET",dataType:"script",cache:!0,async:!1,global:!1,converters:{"text
script":function(){}},dataFilter:function(e){k.globalEval(e,t)}})},k.fn.extend({wrapAll:function(e){var
t;return
this[0]&&(m(e)&&(e=e.call(this[0])),t=k(e,this[0].ownerDocument).eq(0).clone(!0),this[0].parentNode&&t.insertBefore(th
e=this;while(e.firstElementChild)e=e.firstElementChild;return
e}).append(this)),this},wrapInner:function(n){return
m(n)?this.each(function(e){k(this).wrapInner(n.call(this,e))}):this.each(function(){var
e=k(this),t=e.contents();t.length?t.wrapAll(n):e.append(n)})},wrap:function(t){var n=m(t);return
this.each(function(e){k(this).wrapAll(n?t.call(this,e):t)})},unwrap:function(e){return
this.parent(e).not("body").each(function(){k(this).replaceWith(this.childNodes)}),this}}),k.expr.pseudos.hidden=functi
new C.XMLHttpRequest}catch(e){}};var
Ut={0:200,1223:204},Xt=k.ajaxSettings.xhr();y.cors=!!Xt&&"withCredentials"in
Xt,y.ajax=Xt=!!Xt,k.ajaxTransport(function(i){var
o,a;if(y.cors||Xt&&!i.crossDomain)return{send:function(e,t){var
n,r=i.xhr();if(r.open(i.type,i.url,i.async,i.username,i.password),i.xhrFields)for(n in
i.xhrFields)r[n]=i.xhrFields[n];for(n in
i.mimeType&&r.overrideMimeType&&r.overrideMimeType(i.mimeType),i.crossDomain||e["X-Requested-With"]||(e["X-Requested-W
function(){o&&(o=a=r.onload=r.onerror=r.onabort=r.ontimeout=r.onreadystatechange=null,"abort"===e?r.abort():"error"===e
r.status?t(0,"error"):t(r.status,r.statusText):t(Ut[r.status]||r.status,r.statusText,"text"!==(r.responseType||"text")
r.responseText?{binary:r.response}:{text:r.responseText},r.getAllResponseHeaders()))}},r.onload=o(),a=r.onerror=r.onti
0!==r.onabort?r.onabort=a:r.onreadystatechange=function(){4===r.readyState&&C.setTimeout(function(){o&&a()})},o=o("abo
e}},abort:function(){o&&o()}}}),k.ajaxPrefilter(function(e){e.crossDomain&&(e.contents.script=!1)}),k.ajaxSetup({accept
application/javascript, application/ecmascript,
application/x-ecmascript"},contents:{script:/\b(?:java|ecma)script\b/},converters:{"text
```

```
script":function(e){return k.globalEval(e),e}}}),k.ajaxPrefilter("script",function(e){void
0===e.cache&&(e.cache=!1),e.crossDomain&&(e.type="GET")}),k.ajaxTransport("script",function(n){var
r,i;if(n.crossDomain||n.scriptAttrs)return{send:function(e,t){r=k("<script>").attr(n.scriptAttrs||{}).prop({charset:n.s
error",i=function(e){r.remove(),i=null,e&&t("error"===e.type?404:200,e.type)}),E.head.appendChild(r[0])},abort:function
Vt,Gt=[],Yt=/(=)\?(?=&|$)|\?\?/;k.ajaxSetup({jsonp:"callback",jsonpCallback:function(){var
e=Gt.pop()||k.expando+"_"+kt++;return this[e]=!0,e}}),k.ajaxPrefilter("json
jsonp",function(e,t,n){var r,i,o,a=!1!==e.jsonp&&(Yt.test(e.url)?"url":"string"==typeof
e.data&&0===(e.contentType||"").indexOf("application/x-www-form-urlencoded")&&Yt.test(e.data)&&"data");if(a||"jsonp"===
r=e.jsonpCallback=m(e.jsonpCallback)?e.jsonpCallback():e.jsonpCallback,a?e[a]=e[a].replace(Yt,"$1"+r):!1!==e.jsonp&&(e
json"]=function(){return o||k.error(r+" was not
called"),o[0]},e.dataTypes[0]="json",i=C[r],C[r]=function(){o=arguments},n.always(function(){void
0===i?k(C).removeProp(r):C[r]=i,e[r]&&(e.jsonpCallback=t.jsonpCallback,Gt.push(r)),o&&m(i)&&i(o[0]),o=i=void
0}),"script"}),y.createHTMLDocument=((Vt=E.implementation.createHTMLDocument("").body).innerHTML="<form></form><form></
e?[]:("boolean"==typeof
t&&(n=t,t=!1),t||(y.createHTMLDocument?((r=(t=E.implementation.createHTMLDocument("")).createElement("base")).href=E.lo
r,i,o},k.fn.load=function(e,t,n){var r,i,o,a=this,s=e.indexOf("
");return-1<s&&(r=mt(e.slice(s)),e=e.slice(0,s)),m(t)?(n=t,t=void 0):t&&"object"==typeof
t&&(i="POST"),0<a.length&&k.ajax({url:e,type:i||"GET",dataType:"html",data:t}).done(function(e){o=arguments,a.html(r?k
this.on(t,e)}}),k.expr.pseudos.animated=function(t){return k.grep(k.timers,function(e){return
t===e.elem}).length},k.offset={setOffset:function(e,t,n){var
r,i,o,a,s,u,l=k.css(e,"position"),c=k(e),f={};"static"===l&&(e.style.position="relative"),s=c.offset(),o=k.css(e,"top")
t?t.using.call(e,f):c.css(f)}},k.fn.extend({offset:function(t){if(arguments.length)return void
0===t?this:this.each(function(e){k.offset.setOffset(this,t,e)});var e,n,r=this[0];return
r?r.getClientRects().length?(e=r.getBoundingClientRect(),n=r.ownerDocument.defaultView,{top:e.top+n.pageYOffset,left:e
0},position:function(){if(this[0]){var
e,t,n,r=this[0],i={top:0,left:0};if("fixed"===k.css(r,"position"))t=r.getBoundingClientRect();else{t=this.offset(),n=r
this.map(function(){var
e=this.offsetParent;while(e&&"static"===k.css(e,"position"))e=e.offsetParent;return
e||ie})}}),k.each({scrollLeft:"pageXOffset",scrollTop:"pageYOffset"},function(t,i){var
o="pageYOffset"===i;k.fn[t]=function(e){return _(this,function(e,t,n){var
r;if(x(e)?r=e:9===e.nodeType&&(r=e.defaultView),void 0===n)return
r?r[i]:e[t];r?r.scrollTo(o?r.pageXOffset:n,o?n:r.pageYOffset):e[t]=n},t,e,arguments.length)}},k.each(["top","left"],fu
t=_e(e,n),$e.test(t)?k(e).position()[n]+"px":t})}},k.each({Height:"height",Width:"width"},function(a,s){k.each({padding
n=arguments.length&&(r||"boolean"!=typeof e),i=r||(!0===e||!0===t?"margin":"border");return
_(this,function(e,t,n){var r;return
x(e)?0===o.indexOf("outer")?e["inner"+a]:e.document.documentElement["client"+a]:9===e.nodeType?(r=e.documentElement,Mat
0===n?k.css(e,t,i):k.style(e,t,n,i)},s,n?e:void 0,n)}})}),k.each("blur focus focusin focusout resize
scroll click dblclick mousedown mouseup mousemove mouseover mouseout mouseenter mouseleave change
select submit keydown keypress keyup contextmenu".split("
"),function(e,n){k.fn[n]=function(e,t){return
0<arguments.length?this.on(n,null,e,t):this.trigger(n)}}),k.fn.extend({hover:function(e,t){return
this.mouseenter(e).mouseleave(t||e)}}),k.fn.extend({bind:function(e,t,n){return
this.on(e,null,t,n)},unbind:function(e,t){return
this.off(e,null,t)},delegate:function(e,t,n,r){return
this.on(t,e,n,r)},undelegate:function(e,t,n){return
1===arguments.length?this.off(e,"**"):this.off(t,e||"**",n)}}),k.proxy=function(e,t){var
n,r,i;if("string"==typeof t&&(n=e[t],t=e,e=n),m(e))return r=s.call(arguments,2),(i=function(){return
e.apply(t||this,r.concat(s.call(arguments)))}).guid=e.guid=e.guid||k.guid++,i},k.holdReady=function(e){e?k.readyWait++
t=k.type(e);return("number"===t||"string"===t)&&!isNaN(e-parseFloat(e))},"function"==typeof
define&&define.amd&&define("jquery",[],function(){return k});var Qt=C.jQuery,Jt=C.$;return
k.noConflict=function(e){return
C.$===k&&(C.$=Jt),e&&C.jQuery===k&&(C.jQuery=Qt),k},e||(C.jQuery=C.$=k),k});
/*! jQuery UI - v1.12.1 - 2019-01-27
* http://jqueryui.com
* Includes: widget.js, position.js, data.js, disable-selection.js, focusable.js, form-reset-mixin.js,
jquery-1-7.js, keycode.js, labels.js, scroll-parent.js, tabbable.js, unique-id.js,
widgets/resizable.js, widgets/mouse.js
* Copyright jQuery Foundation and other contributors; Licensed MIT */

(function(t){"function"==typeof
define&&define.amd?define(["jquery"],t):t(jQuery)})(function(t){function e(t){for(var
e=t.css("visibility");"inherit"===e;)t=t.parent(),e=t.css("visibility");return"hidden"!==e}t.ui=t.ui||{},t.ui.version="
i=0,s=Array.prototype.slice;t.cleanData=function(e){return function(i){var
s,n,o;for(o=0;null!=(n=i[o]);o++)try{s=t._data(n,"events"),s&&s.remove&&t(n).triggerHandler("remove")}catch(a){}e(i)}}
n,o,a,r={},h=e.split(".")[0];e=e.split(".")[1];var l=h+"-"+e;return
s||(s=i,i=t.Widget),t.isArray(s)&&(s=t.extend.apply(null,[{}].concat(s))),t.expr[":"][l.toLowerCase()]=function(e){retu
this._createWidget?(arguments.length&&this._createWidget(t,e),void 0):new
o(t,e)},t.extend(o,n,{version:s.version,_proto:t.extend({},s),_childConstructors:[]}),a=new
i,a.options=t.widget.extend({},a.options),t.each(s,function(e,s){return
t.isFunction(s)?(r[e]=function(){function t(){return i.prototype[e].apply(this,arguments)}function
n(t){return i.prototype[e].apply(this,t)}return function(){var
e,i=this._super,o=this._superApply;return
this._super=t,this._superApply=n,e=s.apply(this,arguments),this._super=i,this._superApply=o,e}}(),void
0):(r[e]=s,void
0)}),o.prototype=t.widget.extend(a,{widgetEventPrefix:n?a.widgetEventPrefix||e:e},r,{constructor:o,namespace:h,widgetNa
s=i.prototype;t.widget(s.namespace+"."+s.widgetName,o,i._proto)}),delete
n._childConstructors):i._childConstructors.push(o),t.widget.bridge(e,o),o},t.widget.extend=function(e){for(var
i,n,o=s.call(arguments,1),a=0,r=o.length;r>a;a++)for(i in o[a])n=o[a][i],o[a].hasOwnProperty(i)&&void
0!==n&&(e[i]=t.isPlainObject(n)?t.isPlainObject(e[i])?t.widget.extend({},e[i],n):t.widget.extend({},n):n);return
e},t.widget.bridge=function(e,i){var n=i.prototype.widgetFullName||e;t.fn[e]=function(o){var
a="string"==typeof o,r=s.call(arguments,1),h=this;return
a?this.length||"instance"!==o?this.each(function(){var
i,s=t.data(this,n);return"instance"===o?(h=s,!1):s?t.isFunction(s[o])&&"_"!==o.charAt(0)?(i=s[o].apply(s,r),i!==s&&void
0!==i?(h=i&&i.jquery?h.pushStack(i.get()):i,!1):void 0):t.error("no such method '"+o+"' for "+e+"
widget instance"):t.error("cannot call methods on "+e+" prior to initialization; "+"attempted to call
method '"+o+"'")}):h=void
0:(r.length&&(o=t.widget.extend.apply(null,[o].concat(r))),this.each(function(){var
```

```
e=t.data(this,n);e?(e.option(o||{}),e._init&&e._init()):t.data(this,n,new
i(o,this))})),h}},t.Widget=function(){},t.Widget._childConstructors=[],t.Widget.prototype={widgetName:"widget",widgetEv
e=this;this._destroy(),t.each(this.classesElementLookup,function(t,i){e._removeClass(i,t)}),this.element.off(this.event
this.element},option:function(e,i){var s,n,o,a=e;if(0===arguments.length)return
t.widget.extend({},this.options);if("string"==typeof
e)if(a={},s=e.split("."),e=s.shift(),s.length){for(n=a[e]=t.widget.extend({},this.options[e]),o=0;s.length-1>o;o++)n[s
void 0===n[e]?null:n[e];n[e]=i}else{if(1===arguments.length)return void
0===this.options[e]?null:this.options[e];a[e]=i}return
this._setOptions(a),this},_setOptions:function(t){var e;for(e in t)this._setOption(e,t[e]);return
this},_setOption:function(t,e){return"classes"===t&&this._setOptionClasses(e),this.options[t]=e,"disabled"===t&&this._s
i,s,n;for(i in
e)n=this.classesElementLookup[i],e[i]!==this.options.classes[i]&&n&&n.length&&(s=t(n.get()),this._removeClass(n,i),s.ad
this._setOptions({disabled:!1})},disable:function(){return
this._setOptions({disabled:!0})},_classes:function(e){function i(i,o){var
a,r;for(r=0;i.length>r;r++)a=n.classesElementLookup[i[r]]||t(),a=e.add?t(t.unique(a.get().concat(e.element.get()))):t(a
s=[],n=this;return
e=t.extend({element:this.element,classes:this.options.classes||{}},e),this._on(e.element,{remove:"_untrackClassesElemen
")},_untrackClassesElement:function(e){var
i=this;t.each(i.classesElementLookup,function(s,n){-1!==t.inArray(e.target,n)&&(i.classesElementLookup[s]=t(n.not(e.tar
this._toggleClass(t,e,i,!1)},_addClass:function(t,e,i){return
this._toggleClass(t,e,i,!0)},_toggleClass:function(t,e,i,s){s="boolean"==typeof s?s:i;var
n="string"==typeof t||null===t,o={extra:n?e:i,keys:n?t:e,element:n?this.element:t,add:s};return
o.element.toggleClass(this._classes(o),s),this},_on:function(e,i,s){var n,o=this;"boolean"!=typeof
e&&(s=i,i=e,e=!1),s?(i=n=t(i),this.bindings=this.bindings.add(i)):(s=i,i=this.element,n=this.widget()),t.each(s,functio
r(){return e||o.options.disabled!==!0&&!t(this).hasClass("ui-state-disabled")?("string"==typeof
a?o[a]:a).apply(o,arguments):void 0}"string"!=typeof a&&(r.guid=a.guid=a.guid||r.guid||t.guid++);var
h=s.match(/^([\w:-]*)\s*(.*)$/),l=h[1]+o.eventNamespace,c=h[2];c?n.on(l,c,r):i.on(l,r)})},_off:function(e,i){i=(i||"")
").join(this.eventNamespace+"
")+this.eventNamespace,e.off(i).off(i),this.bindings=t(this.bindings.not(e).get()),this.focusable=t(this.focusable.not
i(){return("string"==typeof t?s[t]:t).apply(s,arguments)}var s=this;return
setTimeout(i,e||0)},_hoverable:function(e){this.hoverable=this.hoverable.add(e),this._on(e,{mouseenter:function(e){this
n,o,a=this.options[e];if(s=s||{},i=t.Event(i),i.type=(e===this.widgetEventPrefix?e:this.widgetEventPrefix+e).toLowerCas
in o)n in i||(i[n]=o[n]);return
this.element.trigger(i,s),!(t.isFunction(a)&&a.apply(this.element[0],[i].concat(s))===!1||i.isDefaultPrevented())}},t.e
n&&(n={effect:n});var a,r=n?n===!0||"number"==typeof n?i:n.effect||i:e;n=n||{},"number"==typeof
n&&(n={duration:n}),a=!t.isEmptyObject(n),n.complete=o,n.delay&&s.delay(n.delay),a&&t.effects&&t.effects.effect[r]?s[e
e(t,e,i){return[parseFloat(t[0])*(u.test(t[0])?e/100:1),parseFloat(t[1])*(u.test(t[1])?i/100:1)]}function
i(e,i){return parseInt(t.css(e,i),10)||0}function s(e){var i=e[0];return
9===i.nodeType?{width:e.width(),height:e.height(),offset:{top:0,left:0}}:t.isWindow(i)?{width:e.width(),height:e.height
n,o=Math.max,a=Math.abs,r=/left|center|right/,h=/top|center|bottom/,l=/[\+\-]\d+(\.[\d]+)?%?/,c=/^\w+/,u=/%$/,d=t.fn.po
0!==n)return n;var e,i,s=t("<div
style='display:block;position:absolute;width:50px;height:50px;overflow:hidden;'><div
style='height:100px;width:auto;'></div></div>"),o=s.children()[0];return
t("body").append(s),e=o.offsetWidth,s.css("overflow","scroll"),i=o.offsetWidth,e===i&&(i=s[0].clientWidth),s.remove(),n
i=e.isWindow||e.isDocument?"":e.element.css("overflow-x"),s=e.isWindow||e.isDocument?"":e.element.css("overflow-y"),n="
i=t(e||window),s=t.isWindow(i[0]),n=!!i[0]&&9===i[0].nodeType,o=!s&&!n;return{element:i,isWindow:s,isDocument:n,offset
d.apply(this,arguments);n=t.extend({},n);var
u,p,f,m,g,_,v=t(n.of),b=t.position.getWithinInfo(n.within),y=t.position.getScrollInfo(b),w=(n.collision||"flip").split
"),x={};return _=s(v),v[0].preventDefault&&(n.at="left
top"),p=_.width,f=_.height,m=_.offset,g=t.extend({},m),t.each(["my","at"],function(){var
t,e,i=(n[this]||"").split("
");1===i.length&&(i=r.test(i[0])?i.concat(["center"]):h.test(i[0])?["center"].concat(i):["center","center"]),i[0]=r.tes
s,r,h=t(this),l=h.outerWidth(),c=h.outerHeight(),d=i(this,"marginLeft"),_=i(this,"marginTop"),k=l+d+i(this,"marginRight
e=m.left-D.left,i=e+p-l,s=m.top-D.top,r=s+f-c,u={target:{element:v,left:m.left,top:m.top,width:p,height:f},element:{ele
i,s=e.within,n=s.isWindow?s.scrollLeft:s.offset.left,a=s.width,r=t.left-e.collisionPosition.marginLeft,h=n-r,l=r+e.coll
i,s=e.within,n=s.isWindow?s.scrollTop:s.offset.top,a=e.within.height,r=t.top-e.collisionPosition.marginTop,h=n-r,l=r+e
i,s,n=e.within,o=n.offset.left+n.scrollLeft,r=n.width,h=n.isWindow?n.scrollLeft:n.offset.left,l=t.left-e.collisionPosit
i,s,n=e.within,o=n.offset.top+n.scrollTop,r=n.height,h=n.isWindow?n.scrollTop:n.offset.top,l=t.top-e.collisionPosition
function(i){return!!t.data(i,e)})):function(e,i,s){return!!t.data(e,s[3])})})},t.fn.extend({disableSelection:function(){v
t="onselectstart"in document.createElement("div")?"selectstart":"mousedown";return function(){return
this.on(t+".ui-disableSelection",function(t){t.preventDefault()})}}(),enableSelection:function(){return
this.off(".ui-disableSelection")}})},t.ui.focusable=function(i,s){var
n,o,a,r,h,l=i.nodeName.toLowerCase();return"area"===l?(n=i.parentNode,o=n.name,i.href&&o&&"map"===n.nodeName.toLowerCas
t.ui.focusable(e,null!=t.attr(e,"tabindex"))}})},t.ui.focusable,t.fn.form=function(){return"string"==typeof
this[0].form?this.closest("form"):t(this[0].form)},t.ui.formResetMixin={_formResetHandler:function(){var
e=t(this);setTimeout(function(){var
i=e.data("ui-form-reset-instances");t.each(i,function(){this.refresh()})})},_bindFormResetHandler:function(){if(this.fo
t=this.form.data("ui-form-reset-instances")||[];t.length||this.form.on("reset.ui-form-reset",this._formResetHandler),t
e=this.form.data("ui-form-reset-instances");e.splice(t.inArray(this,e),1),e.length?this.form.data("ui-form-reset-instan
s(e,i,s,o){return
t.each(n,function(){i-=parseFloat(t.css(e,"padding"+this))||0,s&&(i-=parseFloat(t.css(e,"border"+this+"Width"))||0),o&&
n="Width"===i?["Left","Right"]:["Top","Bottom"],o=i.toLowerCase(),a={innerWidth:t.fn.innerWidth,innerHeight:t.fn.innerH
void
0===e?a["inner"+i].call(this):this.each(function(){t(this).css(o,s(this,e)+"px")})},t.fn["outer"+i]=function(e,n){retur
e?a["outer"+i].call(this,e):this.each(function(){t(this).css(o,s(this,e,!0,n)+"px")})}}),t.fn.addBack=function(t){retur
this.add(null==t?this.prevObject:this.prevObject.filter(t))},t.ui.keyCode={BACKSPACE:8,COMMA:188,DELETE:46,DOWN:40,END
t=/([!"#$%&'()*+,./:;<=>?@[\]^`{|}~])/g;return function(e){return
e.replace(t,"\\$1")}}(),t.fn.labels=function(){var e,i,s,n,o;return
this[0].labels&&this[0].labels.length?this.pushStack(this[0].labels):(n=this.eq(0).parents("label"),s=this.attr("id"),s
i=this.css("position"),s="absolute"===i,n=e?/(auto|scroll|hidden)/:/(auto|scroll)/,o=this.parents().filter(function(){v
e=t(this);return
s&&"static"===e.css("position")?!1:n.test(e.css("overflow")+e.css("overflow-y")+e.css("overflow-x"))}).eq(0);return"fix
i=t.attr(e,"tabindex"),s=null!=i;return(!s||i>=0)&&t.ui.focusable(e,s)}}),t.fn.extend({uniqueId:function(){var
t=0;return function(){return this.each(function(){this.id||(this.id="ui-id-"+
++t)})}}(),removeUniqueId:function(){return
this.each(function(){/^ui-id-\d+$/.test(this.id)&&t(this).removeAttr("id")})}})},t.ui.ie=!!/msie
[\w.]+/.exec(navigator.userAgent.toLowerCase());var
```

```
n=!1;t(document).on("mouseup",function(){n=!1}),t.widget("ui.mouse",{version:"1.12.1",options:{cancel:"input,
textarea, button, select, option",distance:1,delay:0},_mouseInit:function(){var
e=this;this.element.on("mousedown."+this.widgetName,function(t){return
e._mouseDown(t)}).on("click."+this.widgetName,function(i){return!0===t.data(i.target,e.widgetName+".preventClickEvent")
0}),this.started=!1},_mouseDestroy:function(){this.element.off("."+this.widgetName),this._mouseMoveDelegate&&this.docu
i=this,s=1===e.which,o="string"==typeof
this.options.cancel&&e.target.nodeName?t(e.target).closest(this.options.cancel).length:!1;return
s&&!o&&this._mouseCapture(e)?(this.mouseDelayMet=!this.options.delay,this.mouseDelayMet||(this._mouseDelayTimer=setTim
i._mouseMove(t)},this._mouseUpDelegate=function(t){return
i._mouseUp(t)},this.document.on("mousemove."+this.widgetName,this._mouseMoveDelegate).on("mouseup."+this.widgetName,th
this._mouseUp(e);if(!e.which)if(e.originalEvent.altKey||e.originalEvent.ctrlKey||e.originalEvent.metaKey||e.originalEve
if(!this.ignoreMissingWhich)return
this._mouseUp(e)}return(e.which||e.button)&&(this._mouseMoved=!0),this._mouseStarted?(this._mouseDrag(e),e.preventDefau
this._mouseDelayTimer),this.ignoreMissingWhich=!1,n=!1,e.preventDefault()},_mouseDistanceMet:function(t){return
Math.max(Math.abs(this._mouseDownEvent.pageX-t.pageX),Math.abs(this._mouseDownEvent.pageY-t.pageY))>=this.options.dist
this.mouseDelayMet},_mouseStart:function(){},_mouseDrag:function(){},_mouseStop:function(){},_mouseCapture:function(){
n,o=t.ui[e].prototype;for(n in
s)o.plugins[n]=o.plugins[n]||[],o.plugins[n].push([i,s[n]])},call:function(t,e,i,s){var
n,o=t.plugins[e];if(o&&(s||t.element[0].parentNode&&11!==t.element[0].parentNode.nodeType))for(n=0;o.length>n;n++)t.opt
ui-icon-gripsmall-diagonal-se"},containment:!1,ghost:!1,grid:!1,handles:"e,s,se",helper:!1,maxHeight:null,maxWidth:null
parseFloat(t)||0},_isNumber:function(t){return!isNaN(parseFloat(t))},_hasScroll:function(e,i){if("hidden"===t(e).css("
s=i&&"left"===i?"scrollLeft":"scrollTop",n=!1;return
e[s]>0?!0:(e[s]=1,n=e[s]>0,e[s]=0,n)},_create:function(){var
e,i=this.options,s=this;this._addClass("ui-resizable"),t.extend(this,{_aspectRatio:!!i.aspectRatio,aspectRatio:i.aspect
class='ui-wrapper' style='overflow:
hidden;'></div>").css({position:this.element.css("position"),width:this.element.outerWidth(),height:this.element.outer
e,i=function(e){t(e).removeData("resizable").removeData("ui-resizable").off(".resizable").find(".ui-resizable-handle")
this.elementIsWrapper&&(i(this.element),e=this.element,this.originalElement.css({position:e.css("position"),width:e.out
e,i,s,n,o,a=this.options,r=this;if(this.handles=a.handles||(t(".ui-resizable-handle",this.element).length?{n:".ui-resi
"+n),o.css({zIndex:a.zIndex}),this.handles[e]=".ui-resizable-"+e,this.element.append(o);this._renderAxis=function(e){va
i,s,n,o;e=e||this.element;for(i in
this.handles)this.handles[i].constructor===String?this.handles[i]=this.element.children(this.handles[i]).first().show()
i,s,n=!1;for(i in
this.handles)s=t(this.handles[i])[0],(s===e.target||t.contains(s,e.target))&&(n=!0);return!this.options.disabled&&n},_
i,s,n,o=this.options,a=this.element;return
this.resizing=!0,this._renderProxy(),i=this._num(this.helper.css("left")),s=this._num(this.helper.css("top")),o.contain
o.aspectRatio?o.aspectRatio:this.originalSize.width/this.originalSize.height||1,n=t(".ui-resizable-"+this.axis).css("cu
i,s,n=this.originalMousePosition,o=this.axis,a=e.pageX-n.left||0,r=e.pageY-n.top||0,h=this._change[o];return
this._updatePrevProperties(),h?(i=h.apply(this,[e,a,r]),this._updateVirtualBoundaries(e.shiftKey),(this._aspectRatio||e
i,s,n,o,a,r,h,l=this.options,c=this;return
this._helper&&(i=this._proportionallyResizeElements,s=i.length&&/textarea/i.test(i[0].nodeName),n=s&&this._hasScroll(i
t={};return
this.position.top!==this.prevPosition.top&&(t.top=this.position.top+"px"),this.position.left!==this.prevPosition.left&&
e,i,s,n,o,a=this.options;o={minWidth:this._isNumber(a.minWidth)?a.minWidth:0,maxWidth:this._isNumber(a.maxWidth)?a.maxW
e=this.position,i=this.size,s=this.axis;return
this._isNumber(t.height)?t.width=t.height*this.aspectRatio:this._isNumber(t.width)&&(t.height=t.width/this.aspectRatio)
e=this._vBoundaries,i=this.axis,s=this._isNumber(t.width)&&e.maxWidth&&e.maxWidth<t.width,n=this._isNumber(t.height)&&e
o&&(t.width=e.minWidth),a&&(t.height=e.minHeight),s&&(t.width=e.maxWidth),n&&(t.height=e.maxHeight),o&&l&&(t.left=r-e.m
e=0,i=[],s=[t.css("borderTopWidth"),t.css("borderRightWidth"),t.css("borderBottomWidth"),t.css("borderLeftWidth")],n=[t
t,e=0,i=this.helper||this.element;this._proportionallyResizeElements.length>e;e++)t=this._proportionallyResizeElements
e=this.element,i=this.options;this.elementOffset=e.offset(),this._helper?(this.helper=this.helper||t("<div
style='overflow:hidden;'></div>"),this._addClass(this.helper,this._helper),this.helper.css({width:this.element.outerWid
},_change:{e:function(t,e){return{width:this.originalSize.width+e}},w:function(t,e){var
i=this.originalSize,s=this.originalPosition;return{left:s.left+e,width:i.width-e}},n:function(t,e,i){var
s=this.originalSize,n=this.originalPosition;return{top:n.top+i,height:s.height-i}},s:function(t,e,i){return{height:this
t.extend(this._change.s.apply(this,arguments),this._change.e.apply(this,[e,i,s]))},sw:function(e,i,s){return
t.extend(this._change.s.apply(this,arguments),this._change.w.apply(this,[e,i,s]))},ne:function(e,i,s){return
t.extend(this._change.n.apply(this,arguments),this._change.e.apply(this,[e,i,s]))},nw:function(e,i,s){return
t.extend(this._change.n.apply(this,arguments),this._change.w.apply(this,[e,i,s]))}},_propagate:function(e,i){t.ui.plugi
i=t(this).resizable("instance"),s=i.options,n=i._proportionallyResizeElements,o=n.length&&/textarea/i.test(n[0].nodeNam
s={width:parseFloat(i.element.css("width")),height:parseFloat(i.element.css("height")),top:parseFloat(i.element.css("to
e,i,s,n,o,a,r,h=t(this).resizable("instance"),l=h.options,c=h.element,u=l.containment,d=u instanceof
t?u.get(0):/parent/.test(u)?c.parent().get(0):u;d&&(h.containerElement=t(d),/document/.test(u)||u===document?(h.contain
i,s,n,o,a=t(this).resizable("instance"),r=a.options,h=a.containerOffset,l=a.position,c=a._aspectRatio||e.shiftKey,u={to
e=t(this).resizable("instance"),i=e.options,s=e.containerOffset,n=e.containerPosition,o=e.containerElement,a=t(e.helper
e=t(this).resizable("instance"),i=e.options;t(i.alsoResize).each(function(){var
e=t(this);e.data("ui-resizable-alsoresize",{width:parseFloat(e.width()),height:parseFloat(e.height()),left:parseFloat(e
s=t(this).resizable("instance"),n=s.options,o=s.originalSize,a=s.originalPosition,r={height:s.size.height-o.height||0,w
e=t(this),s=t(this).data("ui-resizable-alsoresize"),n={},o=e.parents(i.originalElement[0]).length?["width","height"]:["
i=(s[e]||0)+(r[e]||0);i&&i>=0&&(n[e]=i||null)}),e.css(n)})},stop:function(){t(this).removeData("ui-resizable-alsoresize
e=t(this).resizable("instance"),i=e.size;e.ghost=e.originalElement.clone(),e.ghost.css({opacity:.25,display:"block",pos
e.options.ghost&&e.ghost.addClass(this.options.ghost),e.ghost.appendTo(e.helper)},resize:function(){var
e=t(this).resizable("instance");e.ghost&&e.ghost.css({position:"relative",height:e.size.height,width:e.size.width})},st
e=t(this).resizable("instance");e.ghost&&e.helper&&e.helper.get(0).removeChild(e.ghost.get(0))}})},t.ui.plugin.add("resi
e,i=t(this).resizable("instance"),s=i.options,n=i.size,o=i.originalSize,a=i.originalPosition,r=i.axis,h="number"==typeo
s.grid?[s.grid,s.grid]:s.grid,l=h[0]||1,c=h[1]||1,u=Math.round((n.width-o.width)/l)*l,d=Math.round((n.height-o.height)/
 * Copyright (c) 2007 Ariel Flesler - aflesler  gmail • com | https://github.com/flesler
 * Licensed under MIT
 * @author Ariel Flesler
 * @version 2.1.2
 */
;(function(f){"use strict";"function"===typeof
define&&define.amd?define(["jquery"],f):"undefined"!==typeof
module&&module.exports?module.exports=f(require("jquery")):f(jQuery)})(function($){"use
strict";function
n(a){return!a.nodeName||-1!==$.inArray(a.nodeName.toLowerCase(),["iframe","#document","html","body"])}function
h(a){return $.isFunction(a)||$.isPlainObject(a)?a:{top:a,left:a}}var
```

```
p=$.scrollTo=function(a,d,b){return
$(window).scrollTo(a,d,b)};p.defaults={axis:"xy",duration:0,limit:!0};$.fn.scrollTo=function(a,d,b){"object"===
typeof d&&(b=d,d=0);"function"===typeof
b&&(b={onAfter:b});"max"===a&&(a=9E9);b=$.extend({},p.defaults,b);d=d||b.duration;var
u=b.queue&&1<b.axis.length;u&&(d/=2);b.offset=h(b.offset);b.over=h(b.over);return
this.each(function(){function k(a){var
k=$.extend({},b,{queue:!0,duration:d,complete:a&&function(){a.call(q,e,b)}});r.animate(f,k)}if(null!==a){var
l=n(this),q=l?this.contentWindow||window:this,r=$(q),e=a,f={},t;switch(typeof e){case "number":case
"string":if(/^([+-]=?)?\d+(\.\d+)?(px|%)?$/.test(e)){e= h(e);break}e=l?$(e):$(e,q);case
"object":if(e.length===0)return;if(e.is||e.style)t=(e=$(e)).offset()}var
v=$.isFunction(b.offset)&&b.offset(q,e)||b.offset;$.each(b.axis.split(""),function(a,c){var
d="x"===c?"Left":"Top",m=d.toLowerCase(),g="scroll"+d,h=r[g](),n=p.max(q,c);t?(f[g]=t[m]+(l?0:h-r.offset()[m]),b.margin
"%"===d.slice(-1)?parseFloat(d)/100*n:d);b.limit&&/^\d+$/.test(f[g])&&(f[g]=0>=f[g]?0:Math.min(f[g],n));!a&&1<b.axis.le
b="x"===d?"Width":"Height",h="scroll"+b;if(!n(a))return a[h]-$(a)[b.toLowerCase()]();var
b="client"+b,k=a.ownerDocument||a.document,l=k.documentElement,k=k.body;return
Math.max(l[h],k[h])-Math.min(l[b],k[b])};$.Tween.propHooks.scrollLeft=$.Tween.propHooks.scrollTop={get:function(a){retu
$(a.elem)[a.prop]()}, set:function(a){var
d=this.get(a);if(a.options.interrupt&&a._last&&a._last!==d)return $(a.elem).stop();var
b=Math.round(a.now);d!==b&&($(a.elem)[a.prop](b),a._last=this.get(a))}};return p});
/*!
PowerTip v1.3.1 (2018-04-15)
https://stevenbenner.github.io/jquery-powertip/
Copyright (c) 2018 Steven Benner (http://stevenbenner.com/).
Released under MIT license.
https://raw.github.com/stevenbenner/jquery-powertip/master/LICENSE.txt
*/
(function(root,factory){if(typeof define=="function"&&define.amd){define(["jquery"],factory)}else
if(typeof
module=="object"&&module.exports){module.exports=factory(require("jquery"))}else{factory(root.jQuery)}})(this,functio
$document=$(document),$window=$(window),$body=$("body");var
DATA_DISPLAYCONTROLLER="displayController",DATA_HASACTIVEHOVER="hasActiveHover",DATA_FORCEDOPEN="forcedOpen",DATA_HASM
session={tooltips:null,isTipOpen:false,isFixedTipOpen:false,isClosing:false,tipOpenImminent:false,activeHover:null,cur
Collision={none:0,top:1,bottom:2,left:4,right:8};$.fn.powerTip=function(opts,arg){var
targetElements=this,options,tipController;if(!targetElements.length){return
targetElements}if($.type(opts)=="string"&&$.powerTip[opts]){return
$.powerTip[opts].call(targetElements,targetElements,arg)}options=$.extend({},$.fn.powerTip.defaults,opts);tipControlle
TooltipController(options);initTracking();targetElements.each(function elementSetup(){var
$this=$(this),dataPowertip=$this.data(DATA_POWERTIP),dataElem=$this.data(DATA_POWERTIPJQ),dataTarget=$this.data(DATA_P
DisplayController($this,options,tipController)})};if(!options.manual){$.each(options.openEvents,function(idx,evt){if($
elementToggle(event){$.powerTip.toggle(this,event)})}else{targetElements.on(evt+EVENT_NAMESPACE,function
elementOpen(event){$.powerTip.show(this,event)})}});$.each(options.closeEvents,function(idx,evt){if($.inArray(evt,opti
elementClose(event){$.powerTip.hide(this,!isMouseEvent(event))})}});targetElements.on("keydown"+EVENT_NAMESPACE,functi
elementKeyDown(event){if(event.keyCode===27){$.powerTip.hide(this,true)}})}return
targetElements};$.fn.powerTip.defaults={fadeInTime:200,fadeOutTime:100,followMouse:false,popupId:"powerTip",popupClass
apiShowTip(element,event){if(isMouseEvent(event)){trackMouse(event);session.previousX=event.pageX;session.previousY=eve
element},reposition:function
apiResetPosition(element){$(element).first().data(DATA_DISPLAYCONTROLLER).resetPosition();return
element},hide:function apiCloseTip(element,immediate){var
displayController;immediate=element?immediate:true;if(element){displayController=$(element).first().data(DATA_DISPLAYC
if(session.activeHover){displayController=session.activeHover.data(DATA_DISPLAYCONTROLLER)}if(displayController){displa
element},toggle:function
apiToggle(element,event){if(session.activeHover&&session.activeHover.is(element)){$.powerTip.hide(element,!isMouseEvent
element}};$.powerTip.showTip=$.powerTip.show;$.powerTip.closeTip=$.powerTip.hide;function
CSSCoordinates(){var
me=this;me.top="auto";me.left="auto";me.right="auto";me.bottom="auto";me.set=function(property,value){if($.isNumeric(va
DisplayController(element,options,tipController){var hoverTimer=null,myCloseDelay=null;function
openTooltip(immediate,forceOpen){cancelTimer();if(!element.data(DATA_HASACTIVEHOVER)){if(!immediate){session.tipOpenImm
intentDelay(){hoverTimer=null;checkForIntent()},options.intentPollInterval)}else{if(forceOpen){element.data(DATA_FORCED
closeTooltip(disableDelay){if(myCloseDelay){myCloseDelay=session.closeDelayTimeout=clearTimeout(myCloseDelay);session.c
closeDelay(){session.closeDelayTimeout=null;tipController.hideTip(element);session.delayInProgress=false;myCloseDelay=n
checkForIntent(){var
xDifference=Math.abs(session.previousX-session.currentX),yDifference=Math.abs(session.previousY-session.currentY),total
cancelTimer(stopClose){hoverTimer=clearTimeout(hoverTimer);if(session.closeDelayTimeout&&myCloseDelay===session.closeDe
cancelClose(){session.closeDelayTimeout=clearTimeout(session.closeDelayTimeout);session.delayInProgress=false}function
closeAnyDelayed(){if(session.delayInProgress&&session.activeHover&&!session.activeHover.is(element)){session.activeHove
repositionTooltip(){tipController.resetPosition(element)}this.show=openTooltip;this.hide=closeTooltip;this.cancel=cance
PlacementCalculator(){function
computePlacementCoords(element,placement,tipWidth,tipHeight,offset){var
placementBase=placement.split("-")[0],coords=new
CSSCoordinates,position;if(isSvgElement(element)){position=getSvgPlacement(element,placementBase)}else{position=getHtml
coords}function getHtmlPlacement(element,placement){var
objectOffset=element.offset(),objectWidth=element.outerWidth(),objectHeight=element.outerHeight(),left,top;switch(place
getSvgPlacement(element,placement){var
svgElement=element.closest("svg")[0],domElement=element[0],point=svgElement.createSVGPoint(),boundingBox=domElement.get
pushPlacement(){placements.push(point.matrixTransform(matrix))}point.x=boundingBox.x;point.y=boundingBox.y;pushPlacemen
TooltipController(options){var placementCalculator=new
PlacementCalculator,tipElement=$("#"+options.popupId);if(tipElement.length===0){tipElement=$("<div/>",{id:options.popup
beginShowTip(element){element.data(DATA_HASACTIVEHOVER,true);tipElement.queue(function
queueTipInit(next){showTip(element);next()})}function showTip(element){var
tipContent;if(!element.data(DATA_HASACTIVEHOVER)){return}if(session.isTipOpen){if(!session.isClosing){hideTip(session.a
queueTipAgain(next){showTip(element);next()});return}element.trigger("powerTipPreRender");tipContent=getTooltipContent
documentClick(event){var
target=event.target;if(target!==element[0]){if(options.mouseOnToPopup){if(target!==tipElement[0]&&!$.contains(tipElemen
tipMouseEnter(){if(session.activeHover){session.activeHover.data(DATA_DISPLAYCONTROLLER).cancel()}});tipElement.on("mou
tipMouseLeave(){if(session.activeHover){session.activeHover.data(DATA_DISPLAYCONTROLLER).hide()}})}tipElement.fadeIn(op
fadeInCallback(){if(!session.desyncTimeout){session.desyncTimeout=setInterval(closeDesyncedTip,500)}element.trigger("po
hideTip(element){session.isClosing=true;session.isTipOpen=false;session.desyncTimeout=clearInterval(session.desyncTime
```

```
fadeOutCallback(){var coords=new
CSSCoordinates;session.activeHover=null;session.isClosing=false;session.isFixedTipOpen=false;tipElement.removeClass();c
positionTipOnCursor(){var
tipWidth,tipHeight,coords,collisions,collisionCount;if(!session.isFixedTipOpen&&(session.isTipOpen||session.tipOpenImm
CSSCoordinates;coords.set("top",session.currentY+options.offset);coords.set("left",session.currentX+options.offset);col
if(collisions===Collision.bottom){coords.set("top",session.scrollTop+session.windowHeight-tipHeight)}}else{coords.set("
positionTipOnElement(element){var
priorityList,finalPlacement;if(options.smartPlacement||options.followMouse&&element.data(DATA_FORCEDOPEN)){priorityList
collisions=getViewportCollisions(placeTooltip(element,pos),tipElement.outerWidth(),tipElement.outerHeight());finalPlace
collisions!==Collision.none})}else{placeTooltip(element,options.placement);finalPlacement=options.placement}tipElement
nw sw e ne se n s w se-alt sw-alt ne-alt nw-alt");tipElement.addClass(finalPlacement)}function
placeTooltip(element,placement){var iterationCount=0,tipWidth,tipHeight,coords=new
CSSCoordinates;coords.set("top",0);coords.set("left",0);tipElement.css(coords);do{tipWidth=tipElement.outerWidth();tipH
coords}function closeDesyncedTip(){var
isDesynced=false,hasDesyncableCloseEvent=$.grep(["mouseleave","mouseout","blur","focusout"],function(eventType){return
$.inArray(eventType,options.closeEvents)!==-1}).length>0;if(session.isTipOpen&&!session.isClosing&&!session.delayInProc
if(!isMouseOver(session.activeHover)&&!session.activeHover.is(":focus")&&!session.activeHover.data(DATA_FORCEDOPEN)){i
isSvgElement(element){return Boolean(window.SVGElement&&element[0]instanceof SVGElement)}function
isMouseEvent(event){return Boolean(event&&$.inArray(event.type,MOUSE_EVENTS)>-1&&typeof
event.pageX=="number")}function
initTracking(){if(!session.mouseTrackingActive){session.mouseTrackingActive=true;getViewportDimensions();$(getViewportD
getViewportDimensions(){session.scrollLeft=$window.scrollLeft();session.scrollTop=$window.scrollTop();session.windowWid
trackResize(){session.windowWidth=$window.width();session.windowHeight=$window.height()}function
trackScroll(){var
x=$window.scrollLeft(),y=$window.scrollTop();if(x!==session.scrollLeft){session.currentX+=x-session.scrollLeft;session
trackMouse(event){session.currentX=event.pageX;session.currentY=event.pageY}function
isMouseOver(element){var
elementPosition=element.offset(),elementBox=element[0].getBoundingClientRect(),elementWidth=elementBox.right-elementBo
session.currentX>=elementPosition.left&&session.currentX<=elementPosition.left+elementWidth&&session.currentY>=elementP
getTooltipContent(element){var
tipText=element.data(DATA_POWERTIP),tipObject=element.data(DATA_POWERTIPJQ),tipTarget=element.data(DATA_POWERTIPTARGET)
if(tipObject){if($.isFunction(tipObject)){tipObject=tipObject.call(element[0])}if(tipObject.length>0){content=tipObject
if(tipTarget){targetElement=$("#"+tipTarget);if(targetElement.length>0){content=targetElement.html()}}return
content}function getViewportCollisions(coords,elementWidth,elementHeight){var
viewportTop=session.scrollTop,viewportLeft=session.scrollLeft,viewportBottom=viewportTop+session.windowHeight,viewportR
collisions}function countFlags(value){var count=0;while(value){value&=value-1;count++}return
count}return $.powerTip});/*!
 * jQuery UI Touch Punch 0.2.3
 *
 * Copyright 2011-2014, Dave Furfero
 * Dual licensed under the MIT or GPL Version 2 licenses.
 *
 * Depends:
 *  jquery.ui.widget.js
 *  jquery.ui.mouse.js
 */
!function(a){function f(a,b){if(!(a.originalEvent.touches.length>1)){a.preventDefault();var
c=a.originalEvent.changedTouches[0],d=document.createEvent("MouseEvents");d.initMouseEvent(b,!0,!0,window,1,c.screenX,c
document,a.support.touch){var
e,b=a.ui.mouse.prototype,c=b._mouseInit,d=b._mouseDestroy;b._touchStart=function(a){var
b=this;!e&&b._mouseCapture(a.originalEvent.changedTouches[0])&&(e=!0,b._touchMoved=!1,f(a,"mouseover"),f(a,"mousemove")
b=this;b.element.bind({touchstart:a.proxy(b,"_touchStart"),touchmove:a.proxy(b,"_touchMove"),touchend:a.proxy(b,"_touch
b=this;b.element.unbind({touchstart:a.proxy(b,"_touchStart"),touchmove:a.proxy(b,"_touchMove"),touchend:a.proxy(b,"_tou
SmartMenus jQuery Plugin - v1.1.0 - September 17, 2017
 * http://www.smartmenus.org/
 * Copyright Vasil Dinkov, Vadikom Web Ltd. http://vadikom.com; Licensed MIT
 */(function(t){"function"==typeof define&&define.amd?define(["jquery"],t):"object"==typeof
module&&"object"==typeof
module.exports?module.exports=t(require("jquery")):t(jQuery)})(function($){function
initMouseDetection(t){var
e=".smartmenus_mouse";if(mouseDetectionEnabled||t)mouseDetectionEnabled&&t&&($(document).off(e),mouseDetectionEnabled=
i=!0,s=null,o={mousemove:function(t){var e={x:t.pageX,y:t.pageY,timeStamp:(new
Date).getTime()};if(s){var
o=Math.abs(s.x-e.x),a=Math.abs(s.y-e.y);if((o>0||a>0)&&2>=o&&2>=a&&300>=e.timeStamp-s.timeStamp&&(mouse=!0,i)){var
n=$(t.target).closest("a");n.is("a")&&$.each(menuTrees,function(){return
$.contains(this.$root[0],n[0])?(this.itemEnter({currentTarget:n[0]}),!1):void
0}),i=!1}}s=e}};o[touchEvents?"touchstart":"pointerover pointermove pointerout MSPointerOver
MSPointerMove
MSPointerOut"]=function(t){isTouchEvent(t.originalEvent)&&(mouse=!1)},$(document).on(getEventsNS(o,e)),mouseDetectionEn
isTouchEvent(t){return!/^(4|mouse)$/.test(t.pointerType)}function getEventsNS(t,e){e||(e="");var
i={};for(var s in t)i[s.split(" ").join(e+" ")+e]=t[s];return i}var
menuTrees=[],mouse=!1,touchEvents="ontouchstart"in
window,mouseDetectionEnabled=!1,requestAnimationFrame=window.requestAnimationFrame||function(t){return
setTimeout(t,1e3/60)},cancelAnimationFrame=window.cancelAnimationFrame||function(t){clearTimeout(t)},canAnimate=!!$.fn
$.SmartMenus=function(t,e){this.$root=$(t),this.opts=e,this.rootId="",this.accessIdPrefix="",this.$subArrow=null,this.a
t.style||"webkitPerspective"in
t.style,this.wasCollapsible=!1,this.init()},$.extend($.SmartMenus,{hideAll:function(){$.each(menuTrees,function(){this
e=this;if(!t){menuTrees.push(this),this.rootId=((new
Date).getTime()+Math.random()+"").replace(/\D/g,""),this.accessIdPrefix="sm-"+this.rootId+"-",this.$root.hasClass("sm-
i=".smartmenus";this.$root.data("smartmenus",this).attr("data-smartmenus-id",this.rootId).dataSM("level",1).on(getEvent
focusin":$.proxy(this.rootOver,this),"mouseout
focusout":$.proxy(this.rootOut,this),keydown:$.proxy(this.rootKeyDown,this)},i)).on(getEventsNS({mouseenter:$.proxy(thi
orientationchange":$.proxy(this.winResize,this)},i)),this.opts.subIndicators&&(this.$subArrow=$("<span/>").addClass("su
s=/(index|default)\.[^#\?\/]*/i,o=/#.*/,a=window.location.href.replace(s,""),n=a.replace(o,"");this.$root.find("a").eac
t=this.href.replace(s,""),i=$(this);(t==a||t==n)&&(i.addClass("current"),e.opts.markCurrentTree&&i.parentsUntil("[data-
e=".smartmenus";this.$root.removeData("smartmenus").removeAttr("data-smartmenus-id").removeDataSM("level").off(e),e+=th
i=this;this.$root.find("ul").each(function(){var
```

```
t=$(this);t.dataSM("scroll-arrows")&&t.dataSM("scroll-arrows").remove(),t.dataSM("shown-before")&&((i.opts.subMenusMinW
t=$(this);0==t.attr("id").indexOf(i.accessIdPrefix)&&t.removeAttr("id")}).removeClass("has-submenu").removeDataSM("sub"
e=this.$root.offset();this.$disableOverlay=$('<div
class="sm-jquery-disable-overlay"/>').css({position:"absolute",top:e.top,left:e.left,width:this.$root.outerWidth(),heig
this.$touchScrollingSub?(this.$touchScrollingSub=null,void
0):((this.visibleSubMenus.length&&!$.contains(this.$root[0],t.target)||$(t.target).closest("a").length)&&this.menuHideA
0)},docTouchEnd:function(){if(this.lastTouch){if(!(!this.visibleSubMenus.length||void
0!==this.lastTouch.x2&&this.lastTouch.x1!=this.lastTouch.x2||void
0!==this.lastTouch.y2&&this.lastTouch.y1!=this.lastTouch.y2||this.lastTouch.target&&$.contains(this.$root[0],this.lastT
t=this;this.hideTimeout=setTimeout(function(){t.menuHideAll()},350)}this.lastTouch=null}},docTouchMove:function(t){if(t
e=t.originalEvent.touches[0];this.lastTouch.x2=e.pageX,this.lastTouch.y2=e.pageY}},docTouchStart:function(t){var
e=t.originalEvent.touches[0];this.lastTouch={x1:e.pageX,y1:e.pageY,target:e.target}},enable:function(){this.disabled&&
e=$(t).closest("ul");e.dataSM("in-mega");)e=e.parent().closest("ul");return
e[0]||null},getHeight:function(t){return this.getOffset(t,!0)},getOffset:function(t,e){var
i;"none"==t.css("display")&&(i={position:t[0].style.position,visibility:t[0].style.visibility},t.css({position:"absolut
s=t[0].getBoundingClientRect&&t[0].getBoundingClientRect(),o=s&&(e?s.height||s.bottom-s.top:s.width||s.right-s.left);re
o||0===o||(o=e?t[0].offsetHeight:t[0].offsetWidth),i&&t.hide().css(i),o},getStartZIndex:function(t){var
e=parseInt(this[t?"$root":"$firstSub"].css("z-index"));return!t&&isNaN(e)&&(e=parseInt(this.$root.css("z-index"))),isNa
t.touches&&t.touches[0]||t.changedTouches&&t.changedTouches[0]||t},getViewport:function(t){var
e=t?"Height":"Width",i=document.documentElement["client"+e],s=window["inner"+e];return
s&&(i=Math.min(i,s)),i},getViewportHeight:function(){return
this.getViewport(!0)},getViewportWidth:function(){return
this.getViewport()},getWidth:function(t){return
this.getOffset(t)},handleEvents:function(){return!this.disabled&&this.isCSSOn()},handleItemEvents:function(t){return
this.handleEvents()&&!this.isLinkInMegaMenu(t)},isCollapsible:function(){return"static"==this.$firstSub.css("position")
t="fixed"==this.$root.css("position");return
t||this.$root.parentsUntil("body").each(function(){return"fixed"==$(this).css("position")?(t=!0,!1):void
0})},isLinkInMegaMenu:function(t){return
$(this.getClosestMenu(t[0])).hasClass("mega-menu")},isTouchMode:function(){return!mouse||this.opts.noMouseOver||this.is
i=t.closest("ul"),s=i.dataSM("level");if(s>1&&(!this.activatedItems[s-2]||this.activatedItems[s-2][0]!=i.dataSM("parent
o=this;$(i.parentsUntil("[data-smartmenus-id]","ul").get().reverse()).add(i).each(function(){o.itemActivate($(this).dat
a=t.dataSM("sub");a&&(this.isTouchMode()||!this.opts.showOnClick||this.clickActivated)&&this.menuShow(a)}},itemBlur:fun
e=$(t.currentTarget);this.handleItemEvents(e)&&this.$root.triggerHandler("blur.smapi",e[0])},itemClick:function(t){var
e=$(t.currentTarget);if(this.handleItemEvents(e)){if(this.$touchScrollingSub&&this.$touchScrollingSub[0]==e.closest("ul
this.$touchScrollingSub=null,t.stopPropagation(),!1;if(this.$root.triggerHandler("click.smapi",e[0])===!1)return!1;var
i=$(t.target).is(".sub-arrow"),s=e.dataSM("sub"),o=s?2==s.dataSM("level"):!1,a=this.isCollapsible(),n=/toggle$/.test(th
this.focusActivated=!0,!1}else if(a&&(n||i))return
this.itemActivate(e,h),this.menuHide(s),n&&(this.focusActivated=!1),!1;return
this.opts.showOnClick&&o||e.hasClass("disabled")||this.$root.triggerHandler("select.smapi",e[0])===!1?!1:void
0}},itemDown:function(t){var
e=$(t.currentTarget);this.handleItemEvents(e)&&e.dataSM("mousedown",!0)},itemEnter:function(t){var
e=$(t.currentTarget);if(this.handleItemEvents(e)){if(!this.isTouchMode()){this.showTimeout&&(clearTimeout(this.showTime
i=this;this.showTimeout=setTimeout(function(){i.itemActivate(e)},this.opts.showOnClick&&1==e.closest("ul").dataSM("leve
e=$(t.currentTarget);this.handleItemEvents(e)&&(!this.focusActivated||this.isTouchMode()&&e.dataSM("mousedown")||this.a
e=$(t.currentTarget);this.handleItemEvents(e)&&(this.isTouchMode()||(e[0].blur(),this.showTimeout&&(clearTimeout(this.s
e=function(){t.css("z-index","")};this.isCollapsible()?canAnimate&&this.opts.collapsibleHideFunction?this.opts.collapsi
i=t.dataSM("level");this.activatedItems.splice(i-1,1),this.visibleSubMenus.splice($.inArray(t,this.visibleSubMenus),1),
t=this.opts.isPopup?1:0,e=this.visibleSubMenus.length-1;e>=t;e--)this.menuHide(this.visibleSubMenus[e]);this.opts.isPop
e=this.activatedItems.length-1;e>=t;e--){var
i=this.activatedItems[e].dataSM("sub");i&&this.menuHide(i)}},menuInit:function(t){if(!t.dataSM("in-mega")){t.hasClass("
e=2,i=t[0];(i=i.parentNode.parentNode)!=this.$root[0];)e++;var
s=t.prevAll("a").eq(-1);s.length||(s=t.prevAll().find("a").eq(-1)),s.addClass("has-submenu").dataSM("sub",t),t.dataSM("
o=s.attr("id")||this.accessIdPrefix+ ++this.idInc,a=t.attr("id")||this.accessIdPrefix+
++this.idInc;s.attr({id:o,"aria-haspopup":"true","aria-controls":a,"aria-expanded":"false"}),t.attr({id:a,role:"group",
e,i,s=t.dataSM("parent-a"),o=s.closest("li"),a=o.parent(),n=t.dataSM("level"),r=this.getWidth(t),h=this.getHeight(t),u=
y=l+e,I=c+i;if(M&&f>y?e=g?f-y+e:d-w:!M&&y+r>f+b&&(e=g?f+b-r-y+e:w-r),g||(S>h&&I+h>v+S?i+=v+S-h-I:(h>=S||v>I)&&(i+=v-I)
x=this;t.dataSM("scroll-arrows")||t.dataSM("scroll-arrows",$([$('<span class="scroll-up"><span
class="scroll-up-arrow"></span></span>')[0],$('<span class="scroll-down"><span
class="scroll-down-arrow"></span></span>')[0]]).on({mouseenter:function(){t.dataSM("scroll").up=$(this).hasClass("scrol
DOMMouseScroll":function(t){t.preventDefault()}}).insertAfter(t));var
A=".smartmenus_scroll";if(t.dataSM("scroll",{y:this.cssTransforms3d?0:i-m,step:1,itemH:m,subH:h,arrowDownH:this.getHei
DOMMouseScroll":function(e){x.menuScrollMousewheel(t,e)}},A)).dataSM("scroll-arrows").css({top:"auto",left:"0",marginLe
C={};C[touchEvents?"touchstart touchmove touchend":"pointerdown pointermove pointerup MSPointerDown
MSPointerMove
MSPointerUp"]=function(e){x.menuScrollTouch(t,e)},t.css({"touch-action":"none","-ms-touch-action":"none"}).on(getEvents
s,o=t.dataSM("scroll"),a=t.dataSM("scroll-arrows"),n=o.up?o.upEnd:o.downEnd;if(!e&&o.momentum){if(o.momentum*=.92,s=o.m
this.menuScrollStop(t),void 0}else
s=i||(e||!this.opts.scrollAccelerate?this.opts.scrollStep:Math.floor(o.step));var
r=t.dataSM("level");if(this.activatedItems[r-1]&&this.activatedItems[r-1].dataSM("sub")&&this.activatedItems[r-1].dataS
"+o.y+"px, 0)",transform:"translate3d(0, "+o.y+"px,
0)"}:{marginTop:o.y}),mouse&&(o.up&&o.y>o.downEnd||!o.up&&o.y<o.upEnd)&&a.eq(o.up?1:0).show(),o.y==n)mouse&&a.eq(o.up?0
if(!e){this.opts.scrollAccelerate&&o.step<this.opts.scrollStep&&(o.step+=.2);var
h=this;this.scrollTimeout=requestAnimationFrame(function(){h.menuScroll(t)})}},menuScrollMousewheel:function(t,e){if(th
i=(e.wheelDelta||-e.detail)>0;t.dataSM("scroll-arrows").eq(i?0:1).is(":visible")&&(t.dataSM("scroll").up=i,this.menuScr
i=t.dataSM("scroll"),s=$(window).scrollTop()-t.dataSM("parent-a").offset().top-i.itemH;t.dataSM("scroll-arrows").eq(0)
e=t.dataSM("scroll"),i=$(window).scrollTop()-t.dataSM("parent-a").offset().top-e.itemH;this.cssTransforms3d&&(i=-(parse
this.scrollTimeout?(cancelAnimationFrame(this.scrollTimeout),this.scrollTimeout=0,t.dataSM("scroll").step=1,!0):void
0},menuScrollTouch:function(t,e){if(e=e.originalEvent,isTouchEvent(e)){var
i=this.getTouchPoint(e);if(this.getClosestMenu(i.target)==t[0]){var
s=t.dataSM("scroll");if(/(start|down)$/i.test(e.type))this.menuScrollStop(t)?(e.preventDefault(),this.$touchScrollingSu
if(/move$/i.test(e.type)){var o=void 0!==s.touchY?s.touchY:s.touchStartY;if(void
0!==o&&o!=i.pageY){this.$touchScrollingSub=t;var a=i.pageY>o;void
0!==s.up&&s.up!=a&&$.extend(s,{touchStartY:i.pageY,touchStartTime:e.timeStamp}),$.extend(s,{up:a,touchY:i.pageY}),this
void
0!==s.touchY&&((s.momentum=15*Math.pow(Math.abs(i.pageY-s.touchStartY)/(e.timeStamp-s.touchStartTime),2))&&(this.menuSc
s.touchY)}}},menuShow:function(t){if((t.dataSM("beforefirstshowfired")||(t.dataSM("beforefirstshowfired",!0),this.$root
e=t.dataSM("parent-a"),i=this.isCollapsible();if((this.opts.keepHighlighted||i)&&e.addClass("highlighted"),i)t.removeCl
```

```
s=this.getWidth(t);t.css("max-width",this.opts.subMenusMaxWidth),s>this.getWidth(t)&&t.removeClass("sm-nowrap").css("wi
o=function(){t.css("overflow","")};i?canAnimate&&this.opts.collapsibleShowFunction?this.opts.collapsibleShowFunction.ca
e=this;this.hideTimeout=setTimeout(function(){e.menuHideAll()},t?1:this.opts.hideTimeout)},popupShow:function(t,e){if(
alert('SmartMenus jQuery Error:\n\nIf you want to show this menu via the "popupShow" method, set the
isPopup:true option.'),void
0;if(this.hideTimeout&&(clearTimeout(this.hideTimeout),this.hideTimeout=0),this.$root.dataSM("shown-before",!0),canAnim
i=this,s=function(){i.$root.css("overflow","")};canAnimate&&this.opts.showFunction?this.opts.showFunction.call(this,thi
27:var e=this.activatedItems[0];if(e){this.menuHideAll(),e[0].focus();var
i=e.dataSM("sub");i&&this.menuHide(i)}break;case 32:var
s=$(t.target);if(s.is("a")&&this.handleItemEvents(s)){var
i=s.dataSM("sub");i&&!i.is(":visible")&&(this.itemClick({currentTarget:t.target}),t.preventDefault())}}},rootOut:functi
e=this;this.hideTimeout=setTimeout(function(){e.menuHideAll()},this.opts.hideTimeout)}},rootOver:function(t){this.handl
window)||"orientationchange"==t.type){var
e=this.isCollapsible();this.wasCollapsible&&e||(this.activatedItems.length&&this.activatedItems[this.activatedItems.len
if(this.$disableOverlay){var
i=this.$root.offset();this.$disableOverlay.css({top:i.top,left:i.left,width:this.$root.outerWidth(),height:this.$root.o
e?this.data(t+"_smartmenus",e):this.data(t+"_smartmenus")},$.fn.removeDataSM=function(t){return
this.removeData(t+"_smartmenus")},$.fn.smartmenus=function(options){if("string"==typeof options){var
args=arguments,method=options;return Array.prototype.shift.call(args),this.each(function(){var
t=$(this).data("smartmenus");t&&t[method]&&t[method].apply(t,args)})}return this.each(function(){var
dataOpts=$(this).data("sm-options")||null;if(dataOpts)try{dataOpts=eval("("+dataOpts+")")}catch(e){dataOpts=null,alert
jQuery init:\nInvalid "data-sm-options" attribute value syntax.')}new
$.SmartMenus(this,$.extend({},$.fn.smartmenus.defaults,options,dataOpts))})},$.fn.smartmenus.defaults={isPopup:!1,mainM
```

8.37 user/html/jquery.js File Reference

8.38 jquery.js

```
00001 /*! jQuery v3.4.1 | (c) JS Foundation and other contributors | jquery.org/license */
00002 !function(e,t){"use strict";"object"==typeof module&&"object"==typeof
module.exports?module.exports=e.document?t(e,!0):function(e){if(!e.document)throw new Error("jQuery
requires a window with a document");return t(e)}:t(e)}("undefined"!=typeof
window?window:this,function(C,e){"use strict";var
t=[],E=C.document,r=Object.getPrototypeOf,s=t.slice,g=t.concat,u=t.push,i=t.indexOf,n={},o=n.toString,v=n.hasOwnPropert
e&&"number"!=typeof e.nodeType},x=function(e){return
null!=e&&e===e.window},c={type:!0,src:!0,nonce:!0,noModule:!0};function b(e,t,n){var
r,i,o=(n=n||E).createElement("script");if(o.text=e,t)for(r in
c)(i=t[r]||t.getAttribute&&t.getAttribute(r))&&o.setAttribute(r,i);n.head.appendChild(o).parentNode.removeChild(o)}func
w(e){return null==e?e+"":"object"==typeof e||"function"==typeof e?n[o.call(e)]||"object":typeof e}var
f="3.4.1",k=function(e,t){return new k.fn.init(e,t)},p=/^[\s\uFEFF\xA0]+|[\s\uFEFF\xA0]+$/g;function
d(e){var t=!!e&&"length"in
e&&e.length,n=w(e);return!m(e)&&!x(e)&&("array"===n||0===t||"number"==typeof t&&0<t&&t-1 in
e)}k.fn=k.prototype={jquery:f,constructor:k,length:0,toArray:function(){return
s.call(this)},get:function(e){return
null==e?s.call(this):e<0?this[e+this.length]:this[e]},pushStack:function(e){var
t=k.merge(this.constructor(),e);return t.prevObject=this,t},each:function(e){return
k.each(this,e)},map:function(n){return this.pushStack(k.map(this,function(e,t){return
n.call(e,t,e)}))},slice:function(){return
this.pushStack(s.apply(this,arguments))},first:function(){return this.eq(0)},last:function(){return
this.eq(-1)},eq:function(e){var t=this.length,n=+e+(e<0?t:0);return
this.pushStack(0<=n&&n<t?[this[n]]:[])},end:function(){return
this.prevObject||this.constructor()},push:u,sort:t.sort,splice:t.splice},k.extend=k.fn.extend=function(){var
e,t,n,r,i,o,a=arguments[0]||{},s=1,u=arguments.length,l=!1;for("boolean"==typeof
a&&(l=a,a=arguments[s]||{},s++),"object"==typeof
a||m(a)||(a={}),s===u&&(a=this,s--);s<u;s++)if(null!=(e=arguments[s]))for(t in
e)r=e[t],"__proto__"!==t&&a!==r&&(l&&r&&(k.isPlainObject(r)||(i=Array.isArray(r)))?(n=a[t],o=i&&!Array.isArray(n)?[]:i
0!==r&&(a[t]=r));return
a},k.extend({expando:"jQuery"+(f+Math.random()).replace(/\D/g,""),isReady:!0,error:function(e){throw
new Error(e)},noop:function(){},isPlainObject:function(e){var t,n;return!(!e||"[object
Object]"!==o.call(e))&&(!(t=r(e))||"function"==typeof(n=v.call(t,"constructor")&&t.constructor)&&a.call(n)===l)},isEmpt
t;for(t in
e)return!1;return!0},globalEval:function(e,t){b(e,{nonce:t&&t.nonce})},each:function(e,t){var
n,r=0;if(d(e)){for(n=e.length;r<n;r++)if(!1===t.call(e[r],r,e[r]))break}else for(r in
e)if(!1===t.call(e[r],r,e[r]))break;return e},trim:function(e){return
null==e?"":(e+"").replace(p,"")},makeArray:function(e,t){var n=t||[];return
null!=e&&(d(Object(e))?k.merge(n,"string"==typeof
e?[e]:e):u.call(n,e)),n},inArray:function(e,t,n){return
null==t?-1:i.call(t,e,n)},merge:function(e,t){for(var
n=+t.length,r=0,i=e.length;r<n;r++)e[i++]=t[r];return e.length=i,e},grep:function(e,t,n){for(var
r=[],i=0,o=e.length,a=!n;i<o;i++)!t(e[i],i)!==a&&r.push(e[i]);return r},map:function(e,t,n){var
r,i,o=0,a=[];if(d(e))for(r=e.length;o<r;o++)null!=(i=t(e[o],o,n))&&a.push(i);else for(o in
e)null!=(i=t(e[o],o,n))&&a.push(i);return g.apply([],a)},guid:1,support:y}),"function"==typeof
Symbol&&(k.fn[Symbol.iterator]=t[Symbol.iterator]),k.each("Boolean Number String Function Array Date
RegExp Object Error Symbol".split(" "),function(e,t){n["[object "+t+"]"]=t.toLowerCase()});var
h=function(n){var e,d,b,o,i,h,f,g,w,u,l,T,C,a,E,v,s,c,y,k="sizzle"+1*new
Date,m=n.document,S=0,r=0,p=ue(),x=ue(),N=ue(),A=ue(),D=function(e,t){return
e===t&&(l=!0),0},j={}.hasOwnProperty,t=[],q=t.pop,L=t.push,H=t.push,O=t.slice,P=function(e,t){for(var
n=0,r=e.length;n<r;n++)if(e[n]===t)return
n;return-1},R="checked|selected|async|autofocus|autoplay|controls|defer|disabled|hidden|ismap|loop|multiple|open|readon
RegExp(M+"+","g"),B=new RegExp("^"+M+"+|((?:^|[^\\\\])(?:\\\\.)*)"+M+"+$","g"),_=new
RegExp("^"+M+"*,"+M+"*"),z=new RegExp("^"+M+"*([>+~]|"+M+")"+M+"*"),U=new RegExp(M+"|>"),X=new
RegExp($),V=new RegExp("^"+I+"$"),G={ID:new RegExp("^#("+I+")"),CLASS:new
```

```
RegExp("^\\.("+I+")"),TAG:new RegExp("^("+I+"|[*])"),ATTR:new RegExp("^"+W),PSEUDO:new
RegExp("^"+$),CHILD:new
RegExp("^:(only|first|last|nth|nth-last)-(child|of-type)(?:\\("+M+"*(even|odd|(([+-]|)(\\d*)n|)"+M+"*(?:([+-]|)"+M+"*(\
RegExp("^(?:"+R+")$","i"),needsContext:new
RegExp("^"+M+"*[>+~]|:(even|odd|eq|gt|lt|nth|first|last)(?:\\("+M+"*((?:-\\d)?\\d*)"+M+"*\\)|)(?=[^-]|$)","i")},Y=/HTM
\w/,Z=/^(?:#([\w-]+)|(\w+)|\.([\w-]+))$/,ee=/[+~]/,te=new
RegExp("\\\\([\\da-f]{1,6}"+M+"?|("+M+")|.)","ig"),ne=function(e,t,n){var r="0x"+t-65536;return
r!=r||n?t:r<0?String.fromCharCode(r+65536):String.fromCharCode(r»10|55296,1023&r|56320)},re=/([\0-\x1f\x7f]|^-?\d)|^-$
t?"\0"===e?"\ufffd":e.slice(0,-1)+"\\"+e.charCodeAt(e.length-1).toString(16)+"
":"\\"+e},oe=function(){T()},ae=be(function(e){return!0===e.disabled&&"fieldset"===e.nodeName.toLowerCase()},{dir:"pare
n=e.length,r=0;while(e[n++]=t[r++]);e.length=n-1}}}function se(t,e,n,r){var
i,o,a,s,u,l,c,f=e&&e.ownerDocument,p=e?e.nodeType:9;if(n=n||[],"string"!=typeof
t||!t||1!==p&&9!==p&&11!==p)return
n;if(!r&&((e?e.ownerDocument||e:m)!==C&&T(e),e=e||C,E)){if(11!==p&&(u=Z.exec(t)))if(i=u[1]){if(9===p){if(!(a=e.getEleme
n;if(a.id===i)return n.push(a),n}else if(f&&(a=f.getElementById(i))&&y(e,a)&&a.id===i)return
n.push(a),n}else{if(u[2])return
H.apply(n,e.getElementsByTagName(t)),n;if((i=u[3])&&d.getElementsByClassName&&e.getElementsByClassName)return
H.apply(n,e.getElementsByClassName(i)),n}if(d.qsa&&!A[t+"
"]&&(!v||!v.test(t))&&(1!==p||"object"!==e.nodeName.toLowerCase())){if(c=t,f=e,1===p&&U.test(t)){(s=e.getAttribute("id'
"+xe(l[o]);c=l.join(","),f=ee.test(t)&&ye(e.parentNode)||e}try{return
H.apply(n,f.querySelectorAll(c)),n}catch(e){A(t,!0)}finally{s===k&&e.removeAttribute("id")}}}return
g(t.replace(B,"$1"),e,n,r)}function ue(){var r=[];return function e(t,n){return r.push(t+"
")>b.cacheLength&&delete e[r.shift()],e[t+" "]=n}}function le(e){return e[k]=!0,e}function ce(e){var
t=C.createElement("fieldset");try{return!!e(t)}catch(e){return!1}finally{t.parentNode&&t.parentNode.removeChild(t),t=nu
fe(e,t){var n=e.split("|"),r=n.length;while(r--)b.attrHandle[n[r]]=t}function pe(e,t){var
n=t&&e,r=n&&1===e.nodeType&&1===t.nodeType&&e.sourceIndex-t.sourceIndex;if(r)return
r;if(n)while(n=n.nextSibling)if(n===t)return-1;return e?1:-1}function de(t){return
function(e){return"input"===e.nodeName.toLowerCase()&&e.type===t}}function he(n){return
function(e){var t=e.nodeName.toLowerCase();return("input"===t||"button"===t)&&e.type===n}}function
ge(t){return function(e){return"form"in e?e.parentNode&&!1===e.disabled?"label"in e?"label"in
e.parentNode?e.parentNode.disabled===t:e.disabled===t:e.isDisabled===t||e.isDisabled!==!t&&ae(e)===t:e.disabled===t:"la
e&&e.disabled===t}}function ve(a){return le(function(o){return o=+o,le(function(e,t){var
n,r=a([],e.length,o),i=r.length;while(i--)e[n=r[i]]&&(e[n]=!(t[n]=e[n]))})})}function ye(e){return
e&&"undefined"!=typeof e.getElementsByTagName&&e}for(e in d=se.support={},i=se.isXML=function(e){var
t=e.namespaceURI,n=(e.ownerDocument||e).documentElement;return!Y.test(t||n&&n.nodeName||"HTML")},T=se.setDocument=funct
t,n,r=e?e.ownerDocument||e:m;return
r!==C&&9===r.nodeType&&r.documentElement&&(a=(C=r).documentElement,E=!i(C),m!==C&&(n=C.defaultView)&&n.top!==n&&(n.addE
e.className="i",!e.getAttribute("className")}),d.getElementsByTagName=ce(function(e){return
e.appendChild(C.createComment("")),!e.getElementsByTagName("*").length}),d.getElementsByClassName=K.test(C.getElementsE
a.appendChild(e).id=k,!C.getElementsByName||!C.getElementsByName(k).length}),d.getById?(b.filter.ID=function(e){var
t=e.replace(te,ne);return function(e){return
e.getAttribute("id")===t}},b.find.ID=function(e,t){if("undefined"!=typeof t.getElementById&&E){var
n=t.getElementById(e);return n?[n]:[]}}):(b.filter.ID=function(e){var n=e.replace(te,ne);return
function(e){var t="undefined"!=typeof e.getAttributeNode&&e.getAttributeNode("id");return
t&&t.value===n}},b.find.ID=function(e,t){if("undefined"!=typeof t.getElementById&&E){var
n,r,i,o=t.getElementById(e);if(o){if((n=o.getAttributeNode("id"))&&n.value===e)return[o];i=t.getElementsByName(e),r=0;w
t.getElementsByTagName?t.getElementsByTagName(e):d.qsa?t.querySelectorAll(e):void
0}:function(e,t){var
n,r=[],i=0,o=t.getElementsByTagName(e);if("*"===e){while(n=o[i++])1===n.nodeType&&r.push(n);return
r}return o},b.find.CLASS=d.getElementsByClassName&&function(e,t){if("undefined"!=typeof
t.getElementsByClassName&&E)return
t.getElementsByClassName(e)},s=[],v=[],(d.qsa=K.test(C.querySelectorAll))&&(ce(function(e){a.appendChild(e).innerHTML="
id='"+k+"'></a><select id='"+k+"-\r\\' msallowcapture=""><option
selected=""></option></select>",e.querySelectorAll("[msallowcapture^=""]").length&&v.push("[*^$]="+M+"*(?:"|\"\")"),e.que
href=" disabled='disabled'></a><select disabled='disabled'><option/></select>";var
t=C.createElement("input");t.setAttribute("type","hidden"),e.appendChild(t).setAttribute("name","D"),e.querySelectorAll
RegExp(v.join("|")),s=s.length&&new
RegExp(s.join("|")),t=K.test(a.compareDocumentPosition),y=t||K.test(a.contains)?function(e,t){var
n=9===e.nodeType?e.documentElement:e,r=t&&t.parentNode;return
e===r||!(!r||1!==r.nodeType||!(n.contains?n.contains(r):e.compareDocumentPosition&&16&e.compareDocumentPosition(r)))}:
l=!0,0;var n=!e.compareDocumentPosition-!t.compareDocumentPosition;return
n||(1&(n=(e.ownerDocument||e)===(t.ownerDocument||t)?e.compareDocumentPosition(t):1)||!d.sortDetached&&t.compareDocumen
l=!0,0;var n,r=0,i=e.parentNode,o=t.parentNode,a=[e],s=[t];if(!i||!o)return
e===C?-1:t===C?1:i?-1:o?1:u?P(u,e)-P(u,t):0;if(i===o)return
pe(e,t);n=e;while(n=n.parentNode)a.unshift(n);n=t;while(n=n.parentNode)s.unshift(n);while(a[r]===s[r])r++;return
r?pe(a[r],s[r]):a[r]===m?-1:s[r]===m?1:0}),C},se.matches=function(e,t){return
se(e,null,null,t)},se.matchesSelector=function(e,t){if((e.ownerDocument||e)!==C&&T(e),d.matchesSelector&&E&&!A[t+"
"]&&(!s||!s.test(t))&&(!v||!v.test(t)))try{var
n=c.call(e,t);if(n||d.disconnectedMatch||e.document&&11!==e.document.nodeType)return
n}catch(e){A(t,!0)}return
0<se(t,C,null,[e]).length},se.contains=function(e,t){return(e.ownerDocument||e)!==C&&T(e),y(e,t)},se.attr=function(e,t
n=b.attrHandle[t.toLowerCase()],r=n&&j.call(b.attrHandle,t.toLowerCase())?n(e,t,!E):void 0;return
void
0!==r?r:d.attributes||!E?e.getAttribute(t):(r=e.getAttributeNode(t))&&r.specified?r.value:null},se.escape=function(e){
new Error("Syntax error, unrecognized expression: "+e)},se.uniqueSort=function(e){var
t,n=[],r=0,i=0;if(l=!d.detectDuplicates,u=!d.sortStable&&e.slice(0),e.sort(D),l){while(t=e[i++])t===e[i]&&(r=n.push(i)
u=null,e},o=se.getText=function(e){var
t,n="",r=0,i=e.nodeType;if(i){if(1===i||9===i||11===i){if("string"==typeof e.textContent)return
e.textContent;for(e=e.firstChild;e;e=e.nextSibling)n+=o(e)}else if(3===i||4===i)return
e.nodeValue}else while(t=e[r++])n+=o(t);return
n},(b=se.selectors={cacheLength:50,createPseudo:le,match:G,attrHandle:{},find:{},relative:{">":{dir:"parentNode",first
":{dir:"parentNode"},"+":{dir:"previousSibling",first:!0},"~":{dir:"previousSibling"}},preFilter:{ATTR:function(e){retu
e[1]=e[1].replace(te,ne),e[3]=(e[3]||e[4]||e[5]||"").replace(te,ne),"~="===e[2]&&(e[3]=" "+e[3]+"
"),e.slice(0,4)},CHILD:function(e){return
e[1]=e[1].toLowerCase(),"nth"===e[1].slice(0,3)?(e[3]||se.error(e[0]),e[4]=+(e[4]?e[5]+(e[6]||1):2*("even"===e[3]||"odd
t,n=!e[6]&&e[2];return
G.CHILD.test(e[0])?null:(e[3]?e[2]=e[4]||e[5]||"":n&&X.test(n)&&(t=h(n,!0))&&(t=n.indexOf(")",n.length-t)-n.length)&&(e
```

```
t=e.replace(te,ne).toLowerCase();return"*"===e?function(){return!0}:function(e){return
e.nodeName&&e.nodeName.toLowerCase()===t}},CLASS:function(e){var t=p[e+" "];return t||(t=new
RegExp("(^|"+M+")"+e+"("+M+"|$)"))&&p(e,function(e){return t.test("string"==typeof
e.className&&e.className||"undefined"!=typeof
e.getAttribute&&e.getAttribute("class")||"")})},ATTR:function(n,r,i){return function(e){var
t=se.attr(e,n);return
null==t?"!="===r:!r||(t+="","="===r?t===i:"!="===r?t!==i:"^="===r?i&&0===t.indexOf(i):"*="===r?i&&-1<t.indexOf(i):"$="
"+t.replace(F," ")+"
").indexOf(i):"|="===r&&(t===i||t.slice(0,i.length+1)===i+"-"))}},CHILD:function(h,e,t,g,v){var
y="nth"!==h.slice(0,3),m="last"!==h.slice(-4),x="of-type"===e;return
1===g&&0===v?function(e){return!!e.parentNode}:function(e,t,n){var
r,i,o,a,s,u,l=y!==m?"nextSibling":"previousSibling",c=e.parentNode,f=x&&e.nodeName.toLowerCase(),p=!n&&!x,d=!1;if(c){i
if(p&&(d=s=(r=(i=(o=(a=e)[k]||(a[k]={}))[a.uniqueID]||(o[a.uniqueID]={}))[h]||[])[0]===S&&r[1]),!1===d)while(a=++s&&a&&
t,a=b.pseudos[e]||b.setFilters[e.toLowerCase()]||se.error("unsupported pseudo: "+e);return
a[k]?a(o):1<a.length?(t=[e,e,"",o],b.setFilters.hasOwnProperty(e.toLowerCase())?le(function(e,t){var
n,r=a(e,o),i=r.length;while(i--)e[n=P(e,r[i])]=!(t[n]=r[i])}):function(e){return
a(e,0,t)}):a}},pseudos:{not:le(function(e){var r=[],i=[],s=f(e.replace(B,"$1"));return
s[k]?le(function(e,t,n,r){var
i,o=s(e,null,r,[]),a=e.length;while(a--)(i=o[a])&&(e[a]=!(t[a]=i))}):function(e,t,n){return
r[0]=e,s(r,null,n,i),r[0]=null,!i.pop()}}),has:le(function(t){return function(e){return
0<se(t,e).length}}),contains:le(function(t){return
t=t.replace(te,ne),function(e){return-1<(e.textContent||o(e)).indexOf(t)}}),lang:le(function(n){return
V.test(n||"")||se.error("unsupported lang: "+n),n=n.replace(te,ne).toLowerCase(),function(e){var
t;do{if(t=E?e.lang:e.getAttribute("xml:lang")||e.getAttribute("lang"))return(t=t.toLowerCase())===n||0===t.indexOf(n+"
t=n.location&&n.location.hash;return t&&t.slice(1)===e.id},root:function(e){return
e===a},focus:function(e){return
e===C.activeElement&&(!C.hasFocus||C.hasFocus())&&!!(e.type||e.href||~e.tabIndex)},enabled:ge(!1),disabled:ge(!0),chec
t=e.nodeName.toLowerCase();return"input"===t&&!!e.checked||"option"===t&&!!e.selected},selected:function(e){return
e.parentNode&&e.parentNode.selectedIndex,!0===e.selected},empty:function(e){for(e=e.firstChild;e;e=e.nextSibling)if(e.
J.test(e.nodeName)},input:function(e){return Q.test(e.nodeName)},button:function(e){var
t=e.nodeName.toLowerCase();return"input"===t&&"button"===e.type||"button"===t},text:function(e){var
t;return"input"===e.nodeName.toLowerCase()&&"text"===e.type&&(null==(t=e.getAttribute("type"))||"text"===t.toLowerCase
n=0;n<t;n+=2)e.push(n);return e}),odd:ve(function(e,t){for(var n=1;n<t;n+=2)e.push(n);return
e}),lt:ve(function(e,t,n){for(var r=n<0?n+t:t<n?t:n;0<=--r;)e.push(r);return
e}),gt:ve(function(e,t,n){for(var r=n<0?n+t:n;++r<t;)e.push(r);return
e})}}).pseudos.nth=b.pseudos.eq,{radio:!0,checkbox:!0,file:!0,password:!0,image:!0})b.pseudos[e]=de(e);for(e
in{submit:!0,reset:!0})b.pseudos[e]=he(e);function me(){}function xe(e){for(var
t=0,n=e.length,r="";t<n;t++)r+=e[t].value;return r}function be(s,e,t){var
u=e.dir,l=e.next,c=l||u,f=t&&"parentNode"===c,p=r++;return
e.first?function(e,t,n){while(e=e[u])if(1===e.nodeType||f)return
s(e,t,n);return!1}:function(e,t,n){var
r,i,o,a=[S,p];if(n){while(e=e[u])if((1===e.nodeType||f)&&s(e,t,n))return!0}else
while(e=e[u])if(1===e.nodeType||f)if(i=(o=e[k]||(e[k]={}))[e.uniqueID]||(o[e.uniqueID]={}),l&&l===e.nodeName.toLowerCas
a[2]=r[2];if((i[c]=a)[2]=s(e,t,n))return!0}return!1}}function we(i){return
1<i.length?function(e,t,n){var r=i.length;while(r--)if(!i[r](e,t,n))return!1;return!0}:i[0]}function
Te(e,t,n,r,i){for(var
o,a=[],s=0,u=e.length,l=null!=t;s<u;s++)(o=e[s])&&(n&&!n(o,r,i)||(a.push(o),l&&t.push(s)));return
a}function Ce(d,h,g,v,y,e){return v&&!v[k]&&(v=Ce(v)),y&&!y[k]&&(y=Ce(y,e)),le(function(e,t,n,r){var
i,o,a,s=[],u=[],l=t.length,c=e||function(e,t,n){for(var r=0,i=t.length;r<i;r++)se(e,t[r],n);return
n}(h||"*",n.nodeType?[n]:n,[]),f=!d||!e&&h?c:Te(c,s,d,n,r),p=g?y||(e?d:l||v)?[]:t:f;if(g&&g(f,p,n,r),v){i=Te(p,u),v(i,
p=Te(p===t?p.splice(l,p.length):p),y?y(null,t,p,r):H.apply(t,p))}function Ee(e){for(var
i,t,n,r=e.length,o=b.relative[e[0].type],a=o||b.relative[" "],s=o?1:0,u=be(function(e){return
e===i},a,!0),l=be(function(e){return-1<P(i,e)},a,!0),c=[function(e,t,n){var
r=!o&&(n||t!==w)||((i=t).nodeType?u(e,t,n):l(e,t,n));return
i=null,r}];s<r;s++)if(t=b.relative[e[s].type])c=[be(we(c),t)];else{if((t=b.filter[e[s].type].apply(null,e[s].matches))
Ce(1<s&&we(c),1<s&&xe(e.slice(0,s-1).concat({value:"
"===e[s-2].type?"*":""})).replace(B,"$1"),t,s<n&&Ee(e.slice(s,n)),n<r&&Ee(e=e.slice(n)),n<r&&xe(e))}c.push(t)}return
we(c)}return me.prototype=b.filters=b.pseudos,b.setFilters=new me,h=se.tokenize=function(e,t){var
n,r,i,o,a,s,u,l=x[e+" "];if(l)return t?0:l.slice(0);a=e,s=[],u=b.preFilter;while(a){for(o in
n&&!(r=_.exec(a))||(r&&(a=a.slice(r[0].length)||a),s.push(i=[])),n=!1,(r=z.exec(a))&&(n=r.shift(),i.push({value:n,type
")}),a=a.slice(n.length)),b.filter)!(r=G[o].exec(a))||u[o]&&!(r=u[o](r))||(n=r.shift(),i.push({value:n,type:o,matches:
t?a.length:a?se.error(e):x(e,s).slice(0)},f=se.compile=function(e,t){var
n,v,y,m,x,r,i=[],o=[],a=N[e+"
"];if(!a){t||(t=h(e)),n=t.length;while(n--)(a=Ee(t[n]))[k]?i.push(a):o.push(a);(a=N(e,(v=o,m=0<(y=i).length,x=0<v.lengt
o,a,s,u=0,l="0",c=e&&[],f=[],p=w,d=e||x&&b.find.TAG("*",i),h=S+=null==p?1:Math.random()||.1,g=d.length;for(i&&(w=t===C
i&&(S=h,w=p),c},m?le(r):r))).selector=e}return a},g=se.select=function(e,t,n,r){var
i,o,a,s,u,l="function"==typeof
e&&e,c=!r&&h(e=l.selector||e);if(n=n||[],1===c.length){if(2<(o=c[0]=c[0].slice(0)).length&&"ID"===(a=o[0]).type&&9===t
n;l&&(t=t.parentNode),e=e.slice(o.shift().value.length)}i=G.needsContext.test(e)?0:o.length;while(i--){if(a=o[i],b.rela
H.apply(n,r),n;break}}}return(l||f(e,c))(r,t,!E,n,!t||ee.test(e)&&ye(t.parentNode)||t),n},d.sortStable=k.split("").sort
1&e.compareDocumentPosition(C.createElement("fieldset"))}),ce(function(e){return e.innerHTML="<a
href='#'></a>","#"===e.firstChild.getAttribute("href")})||fe("type|href|height|width",function(e,t,n){if(!n)return
e.getAttribute(t,"type"===t.toLowerCase()?1:2)}),d.attributes&&ce(function(e){return
e.innerHTML="<input/>",e.firstChild.setAttribute("value",""),""===e.firstChild.getAttribute("value")})||fe("value",func
e.defaultValue}),ce(function(e){return null==e.getAttribute("disabled")})||fe(R,function(e,t,n){var
r;if(!n)return!0===e[t]?t.toLowerCase():(r=e.getAttributeNode(t))&&r.specified?r.value:null}),se}(C);k.find=h,k.expr=h
T=function(e,t,n){var r=[],i=void
0!==n;while((e=e[t])&&9!==e.nodeType)if(1===e.nodeType){if(i&&k(e).is(n))break;r.push(e)}return
r},S=function(e,t){for(var n=[];e;e=e.nextSibling)1===e.nodeType&&e!==t&&n.push(e);return
n},N=k.expr.match.needsContext;function A(e,t){return
e.nodeName&&e.nodeName.toLowerCase()===t.toLowerCase()}var
D=/^<([a-z][^\/\0>:\x20\t\r\n\f]*)[\x20\t\r\n\f]*\/?>(?:<\/\1>|)$/i;function j(e,n,r){return
m(n)?k.grep(e,function(e,t){return!!n.call(e,t,e)!==r}):n.nodeType?k.grep(e,function(e){return
e===n!==r}):"string"!=typeof
n?k.grep(e,function(e){return-1<i.call(n,e)!==r}):k.filter(n,e,r)}k.filter=function(e,t,n){var
r=t[0];return
n&&(e=":not("+e+")"),1===t.length&&1===r.nodeType?k.find.matchesSelector(r,e)?[r]:[]:k.find.matches(e,k.grep(t,functio
```

```
1===e.nodeType}))},k.fn.extend({find:function(e){var t,n,r=this.length,i=this;if("string"!=typeof
e)return
this.pushStack(k(e).filter(function(){for(t=0;t<r;t++)if(k.contains(i[t],this))return!0}));for(n=this.pushStack([]),t=0
1<r?k.uniqueSort(n):n},filter:function(e){return
this.pushStack(j(this,e||[],!1))},not:function(e){return
this.pushStack(j(this,e||[],!0))},is:function(e){return!!j(this,"string"==typeof
e&&N.test(e)?k(e):e||[],!1).length}});var
q,L=/^(?:\s*(<[\w\W]+>)[^>]*|#([\w-]+))$/;(k.fn.init=function(e,t,n){var r,i;if(!e)return
this;if(n=n||q,"string"==typeof
e){if(!(r="<"===e[0]&&">"===e[e.length-1]&&3<=e.length?[null,e,null]:L.exec(e))||!r[1]&&t)return!t||t.jquery?(t||n).fin
instanceof
k?t[0]:t,k.merge(this,k.parseHTML(r[1],t&&t.nodeType?t.ownerDocument||t:E,!0)),D.test(r[1])&&k.isPlainObject(t))for(r
in t)m(this[r])?this[r](t[r]):this.attr(r,t[r]);return
this}return(i=E.getElementById(r[2]))&&(this[0]=i,this.length=1),this}return
e.nodeType?(this[0]=e,this.length=1,this):m(e)?void
0!==n.ready?n.ready(e):e(k):k.makeArray(e,this)}).prototype=k.fn,q=k(E);var
H=/^(?:parents|prev(?:Until|All))/,O={children:!0,contents:!0,next:!0,prev:!0};function
P(e,t){while((e=e[t])&&1!==e.nodeType);return e}k.fn.extend({has:function(e){var
t=k(e,this),n=t.length;return this.filter(function(){for(var
e=0;e<n;e++)if(k.contains(this,t[e]))return!0})},closest:function(e,t){var
n,r=0,i=this.length,o=[],a="string"!=typeof
e&&k(e);if(!N.test(e))for(;r<i;r++)for(n=this[r];n&&n!==t;n=n.parentNode)if(n.nodeType<11&&(a?-1<a.index(n):1===n.node
this.pushStack(1<o.length?k.uniqueSort(o):o)},index:function(e){return e?"string"==typeof
e?i.call(k(e),this[0]):i.call(this,e.jquery?e[0]:e):this[0]&&this[0].parentNode?this.first().prevAll().length:-1},add:
this.pushStack(k.uniqueSort(k.merge(this.get(),k(e,t))))},addBack:function(e){return
this.add(null==e?this.prevObject:this.prevObject.filter(e))}}),k.each({parent:function(e){var
t=e.parentNode;return t&&11!==t.nodeType?t:null},parents:function(e){return
T(e,"parentNode")},parentsUntil:function(e,t,n){return T(e,"parentNode",n)},next:function(e){return
P(e,"nextSibling")},prev:function(e){return P(e,"previousSibling")},nextAll:function(e){return
T(e,"nextSibling")},prevAll:function(e){return
T(e,"previousSibling")},nextUntil:function(e,t,n){return
T(e,"nextSibling",n)},prevUntil:function(e,t,n){return
T(e,"previousSibling",n)},siblings:function(e){return
S((e.parentNode||{}).firstChild,e)},children:function(e){return
S(e.firstChild)},contents:function(e){return"undefined"!=typeof
e.contentDocument?e.contentDocument:(A(e,"template")&&(e=e.content||e),k.merge([],e.childNodes))}},function(r,i){k.fn[
n=k.map(this,i,e);return"Until"!==r.slice(-5)&&(t=e),t&&"string"==typeof
t&&(n=k.filter(t,n)),1<this.length&&(O[r]||k.uniqueSort(n),H.test(r)&&n.reverse()),this.pushStack(n)}});var
R=/[^\x20\t\r\n\f]+/g;function M(e){return e}function I(e){throw e}function W(e,t,n,r){var
i;try{e&&m(i=e.promise)?i.call(e).done(t).fail(n):e&&m(i=e.then)?i.call(e,t,n):t.apply(void
0,[e].slice(r))}catch(e){n.apply(void 0,[e])}}k.Callbacks=function(r){var e,n;r="string"==typeof
r?(e=r,n={},k.each(e.match(R)||[],function(e,t){n[t]=!0})),n):k.extend({},r);var
i,t,o,a,s=[],u=[],l=-1,c=function(){for(a=a||r.once,o=i=!0;u.length;l=-1){t=u.shift();while(++l<s.length)!1===s[l].appl
s&&(t&&!i&&(l=s.length-1,u.push(t)),function
n(e){k.each(e,function(e,t){m(t)?r.unique&&f.has(t)||s.push(t):t&&t.length&&"string"!==w(t)&&n(t)})}(arguments),t&&!i&&
k.each(arguments,function(e,t){var
n;while(-1<(n=k.inArray(t,s,n)))s.splice(n,1),n<=l&&l--}),this},has:function(e){return
e?-1<k.inArray(e,s):0<s.length},empty:function(){return s&&(s=[]),this},disable:function(){return
a=u=[],s=t="",this},disabled:function(){return!s},lock:function(){return
a=u=[],t||i||(s=t=""),this},locked:function(){return!!a},fireWith:function(e,t){return
a||(t=[e,(t=t||[]).slice?t.slice():t],u.push(t),i||c()),this},fire:function(){return
f.fireWith(this,arguments),this},fired:function(){return!!o}};return
f},k.extend({Deferred:function(e){var
o=[["notify","progress",k.Callbacks("memory"),k.Callbacks("memory"),2],["resolve","done",k.Callbacks("once
memory"),k.Callbacks("once memory"),0,"resolved"],["reject","fail",k.Callbacks("once
memory"),k.Callbacks("once memory"),1,"rejected"]],i="pending",a={state:function(){return
i},always:function(){return s.done(arguments).fail(arguments),this},"catch":function(e){return
a.then(null,e)},pipe:function(){var i=arguments;return
k.Deferred(function(r){k.each(o,function(e,t){var n=m(i[t[4]])&&i[t[4]];s[t[1]](function(){var
e=n&&n.apply(this,arguments);e&&m(e.promise)?e.promise().progress(r.notify).done(r.resolve).fail(r.reject):r[t[0]+"With
u=0;function l(i,o,a,s){return function(){var n=this,r=arguments,e=function(){var
e,t;if(!(i<u)){if((e=a.apply(n,r))===o.promise())throw new TypeError("Thenable
self-resolution");t=e&&("object"==typeof e||"function"==typeof
e)&&e.then,m(t)?s?t.call(e,l(u,o,M,s),l(u,o,I,s)):(u++,t.call(e,l(u,o,M,s),l(u,o,I,s),l(u,o,M,o.notifyWith))):(a!==M&&
0,r=[e]),(s||o.resolveWith)(n,r))}},t=s?e:function(){try{e()}catch(e){k.Deferred.exceptionHook&&k.Deferred.exceptionHoo
0,r=[e]),o.rejectWith(n,r))}};i?t():(k.Deferred.getStackHook&&(t.stackTrace=k.Deferred.getStackHook()),C.setTimeout(t)
k.Deferred(function(e){o[0][3].add(l(0,e,m(r)?r:M,e.notifyWith)),o[1][3].add(l(0,e,m(t)?t:M)),o[2][3].add(l(0,e,m(n)?n
null!=e?k.extend(e,a):a}},s={};return k.each(o,function(e,t){var
n=t[2],r=t[5];a[t[1]]=n.add,r&&n.add(function(){i=r},o[3-e][2].disable,o[3-e][3].disable,o[0][2].lock,o[0][3].lock),n.a
s[t[0]+"With"](this===s?void
0:this,arguments),this},s[t[0]+"With"]=n.fireWith}),a.promise(s),e&&e.call(s,s),s},when:function(e){var
n=arguments.length,t=n,r=Array(t),i=s.call(arguments),o=k.Deferred(),a=function(t){return
function(e){r[t]=this,i[t]=1<arguments.length?s.call(arguments):e,--n||o.resolveWith(r,i)}};if(n<=1&&(W(e,o.done(a(t))
o.then();while(t--)W(i[t],a(t),o.reject);return o.promise()}});var
$=/^(Eval|Internal|Range|Reference|Syntax|Type|URI)Error$/;k.Deferred.exceptionHook=function(e,t){C.console&&C.console
exception: "+e.message,e.stack,t)},k.readyException=function(e){C.setTimeout(function(){throw
e})};var F=k.Deferred();function
B(){E.removeEventListener("DOMContentLoaded",B),C.removeEventListener("load",B),k.ready()}k.fn.ready=function(e){return
F.then(e)["catch"](function(e){k.readyException(e)}),this},k.extend({isReady:!1,readyWait:1,ready:function(e){(!0===e?-
_=function(e,t,n,r,i,o,a){var s=0,u=e.length,l=null==n;if("object"===w(n))for(s in
i=!0,n)_(e,t,s,n[s],!0,o,a);else if(void
0!==r&&(i=!0,m(r)||(a=!0),l&&(a?(t.call(e,r),t=null):(l=t,t=function(e,t,n){return
l.call(k(e),n)})),t))for(;s<u;s++)t(e[s],n,a?r:r.call(e[s],s,t(e[s],n)));return
i?e:l?t.call(e):u?t(e[0],n):o},z=/^-ms-/,U=/-([a-z])/g;function X(e,t){return
t.toUpperCase()}function V(e){return e.replace(z,"ms-").replace(U,X)}var G=function(e){return
1===e.nodeType||9===e.nodeType||!+e.nodeType};function
Y(){this.expando=k.expando+Y.uid++}Y.uid=1,Y.prototype={cache:function(e){var
```

```
t=e[this.expando];return
t||(t={},G(e)&&(e.nodeType?e[this.expando]=t:Object.defineProperty(e,this.expando,{value:t,configurable:!0}))),t},set:
r,i=this.cache(e);if("string"==typeof t)i[V(t)]=n;else for(r in t)i[V(r)]=t[r];return
i},get:function(e,t){return void
0===t?this.cache(e):e[this.expando]&&e[this.expando][V(t)]},access:function(e,t,n){return void
0===t||t&&"string"==typeof t&&void 0===n?this.get(e,t):(this.set(e,t,n),void
0!==n?n:t)},remove:function(e,t){var n,r=e[this.expando];if(void 0!==r){if(void
0!==t){n=(t=Array.isArray(t)?t.map(V):(t=V(t))in r?[t]:t.match(R)||[]).length;while(n--)delete
r[t[n]]}(void 0===t||k.isEmptyObject(r))&&(e.nodeType?e[this.expando]=void 0:delete
e[this.expando])}},hasData:function(e){var t=e[this.expando];return void
0!==t&&!k.isEmptyObject(t)}};var Q=new Y,J=new
Y,K=/^(?:\{[\w\W]*\}|\[[\w\W]*\])$/,Z=/[A-Z]/g;function ee(e,t,n){var r,i;if(void
0===n&&1===e.nodeType)if(r="data-"+t.replace(Z,"-$&").toLowerCase(),"string"==typeof(n=e.getAttribute(r))){try{n="true
n=void 0;return n}k.extend({hasData:function(e){return
J.hasData(e)||Q.hasData(e)},data:function(e,t,n){return
J.access(e,t,n)},removeData:function(e,t){J.remove(e,t)},_data:function(e,t,n){return
Q.access(e,t,n)},_removeData:function(e,t){Q.remove(e,t)}}),k.fn.extend({data:function(n,e){var
t,r,i,o=this[0],a=o&&o.attributes;if(void
0===n){if(this.length&&(i=J.get(o),1===o.nodeType&&!Q.get(o,"hasDataAttrs"))){t=a.length;while(t--)a[t]&&0===(r=a[t].na
i}return"object"==typeof n?this.each(function(){J.set(this,n)}):_(this,function(e){var t;if(o&&void
0===e)return void 0!==(t=J.get(o,n))?t:void 0!==(t=ee(o,n))?t:void
0;this.each(function(){J.set(this,n,e)})},null,e,1<arguments.length,null,!0)},removeData:function(e){return
this.each(function(){J.remove(this,e)})}}),k.extend({queue:function(e,t,n){var r;if(e)return
t=(t||"fx")+"queue",r=Q.get(e,t),n&&(!r||Array.isArray(n)?r=Q.access(e,t,k.makeArray(n)):r.push(n)),r||[]},dequeue:func
n=k.queue(e,t),r=n.length,i=n.shift(),o=k._queueHooks(e,t);"inprogress"===i&&(i=n.shift(),r--),i&&("fx"===t&&n.unshift
o.stop,i.call(e,function(){k.dequeue(e,t)},o)),!r&&o&&o.empty.fire()},_queueHooks:function(e,t){var
n=t+"queueHooks";return Q.get(e,n)||Q.access(e,n,{empty:k.Callbacks("once
memory").add(function(){Q.remove(e,[t+"queue",n])})})}}),k.fn.extend({queue:function(t,n){var
e=2;return"string"!=typeof t&&(n=t,t="fx",e--),arguments.length<e?k.queue(this[0],t):void
0===n?this:this.each(function(){var
e=k.queue(this,t,n);k._queueHooks(this,t),"fx"===t&&"inprogress"!==e[0]&&k.dequeue(this,t)})},dequeue:function(e){retur
this.each(function(){k.dequeue(this,e)})},clearQueue:function(e){return
this.queue(e||"fx",[])},promise:function(e,t){var
n,r=1,i=k.Deferred(),o=this,a=this.length,s=function(){--r||i.resolveWith(o,[o])};"string"!=typeof
e&&(t=e,e=void
0),e=e||"fx";while(a--)(n=Q.get(o[a],e+"queueHooks"))&&n.empty&&(r++,n.empty.add(s));return
s(),i.promise(t)}});var te=/[+-]?(?:\d*\.|)\d+(?:[eE][+-]?\d+|)/.source,ne=new
RegExp("^(?:([+-])=|)("+te+")([a-z%]*)$","i"),re=["Top","Right","Bottom","Left"],ie=E.documentElement,oe=function(e){re
k.contains(e.ownerDocument,e)},ae={composed:!0};ie.getRootNode&&(oe=function(e){return
k.contains(e.ownerDocument,e)||e.getRootNode(ae)===e.ownerDocument});var
se=function(e,t){return"none"===(e=t||e).style.display||""===e.style.display&&oe(e)&&"none"===k.css(e,"display")},ue=fu
i,o,a={};for(o in t)a[o]=e.style[o],e.style[o]=t[o];for(o in
i=n.apply(e,r||[]),t)e.style[o]=a[o];return i};function le(e,t,n,r){var
i,o,a=20,s=r?function(){return r.cur()}:function(){return
k.css(e,t,"")},u=s(),l=n&&n[3]||(k.cssNumber[t]?"":"px"),c=e.nodeType&&(k.cssNumber[t]||"px"!==l&&+u)&&ne.exec(k.css(e,
n&&(c=+c||+u||0,i=n[1]?c+(n[1]+1)*n[2]:+n[2],r&&(r.unit=l,r.start=c,r.end=i)),i}var ce={};function
fe(e,t){for(var
n,r,i,o,a,s,u,l=[],c=0,f=e.length;c<f;c++)(r=e[c]).style&&(n=r.style.display,t?("none"===n&&(l[c]=Q.get(r,"display")||n
0,a=(i=r).ownerDocument,s=i.nodeName,(u=ce[s])||(o=a.body.appendChild(a.createElement(s)),u=k.css(o,"display"),o.parent
e}k.fn.extend({show:function(){return fe(this,!0)},hide:function(){return
fe(this)},toggle:function(e){return"boolean"==typeof
e?e?this.show():this.hide():this.each(function(){se(this)?k(this).show():k(this).hide()})}});var
pe=/^(?:checkbox|radio)$/i,de=/<([a-z][^\/\0>\x20\t\r\n\f]*)/i,he=/^$|^module$|\/(?:java|ecma)script/i,ge={option:[1,"<
multiple='multiple'>","</select>"],thead:[1,"<table>","</table>"],col:[2,"<table><colgroup>","</colgroup></table>"],tr
ve(e,t){var n;return n="undefined"!=typeof
e.getElementsByTagName?e.getElementsByTagName(t||"*"):"undefined"!=typeof
e.querySelectorAll?e.querySelectorAll(t||"*"):[],void 0===t||t&&A(e,t)?k.merge([e],n):n}function
ye(e,t){for(var
n=0,r=e.length;n<r;n++)Q.set(e[n],"globalEval",!t||Q.get(t[n],"globalEval"))}ge.optgroup=ge.option,ge.tbody=ge.tfoot=ge
me,xe,be=/<|&#?\w+;/;function we(e,t,n,r,i){for(var
o,a,s,u,l,c,f=t.createDocumentFragment(),p=[],d=0,h=e.length;d<h;d++)if((o=e[d])||0===o)if("object"===w(o))k.merge(p,o
if(be.test(o)){a=a||f.appendChild(t.createElement("div")),s=(de.exec(o)||["",""])[1].toLowerCase(),u=ge[s]||ge._default
p.push(t.createTextNode(o));f.textContent="",d=0;while(o=p[d++])if(r&&-1<k.inArray(o,r))i&&i.push(o);else
if(l=oe(o),a=ve(f.appendChild(o),"script"),l&&ye(a),n){c=0;while(o=a[c++])he.test(o.type||"")&&n.push(o)}return
f}me=E.createDocumentFragment().appendChild(E.createElement("div")),(xe=E.createElement("input")).setAttribute("type",
Te=/^key/,Ce=/^(?:mouse|pointer|contextmenu|drag|drop)|click/,Ee=/^([^.]*)(?:\.(.+)|)/;function
ke(){return!0}function Se(){return!1}function Ne(e,t){return e===function(){try{return
E.activeElement}catch(e){}}()==("focus"===t)}function Ae(e,t,n,r,i,o){var a,s;if("object"==typeof
t){for(s in"string"!=typeof n&&(r=r||n,n=void 0),t)Ae(e,s,n,r,t[s],o);return
e}if(null==r&&null==i?(i=n,r=n=void 0):null==i&&("string"==typeof n?(i=r,r=void 0):(i=r,r=n,n=void
0)),!1===i)i=Se;else if(!i)return e;return 1===o&&(a=i,(i=function(e){return
k().off(e),a.apply(this,arguments)}).guid=a.guid||(a.guid=k.guid++)),e.each(function(){k.event.add(this,t,i,r,n)})}func
De(e,i,o){o?(Q.set(e,i,!1),k.event.add(e,i,{namespace:!1,handler:function(e){var
t,n,r=Q.get(this,i);if(1&e.isTrigger&&this[i]){if(r.length)(k.event.special[i]||{}).delegateType&&e.stopPropagation();e
if(r=s.call(arguments),Q.set(this,i,r),t=o(this,i),this[i](),r!==(n=Q.get(this,i))||t?Q.set(this,i,!1):n={},r!==n)retur
e.stopImmediatePropagation(),e.preventDefault(),n.value}else
r.length&&(Q.set(this,i,{value:k.event.trigger(k.extend(r[0],k.Event.prototype),r.slice(1),this)}),e.stopImmediateProp
0===Q.get(e,i)&&k.event.add(e,i,ke)}k.event={global:{},add:function(t,e,n,r,i){var
o,a,s,u,l,c,f,p,d,h,g,v=Q.get(t);if(v){n.handler&&(n=(o=n).handler,i=o.selector),i&&k.find.matchesSelector(ie,i),n.guid
k&&k.event.triggered!==e.type?k.event.dispatch.apply(t,arguments):void
0}),l=(e=(e||"").match(R)||[""]).length;while(l--)d=g=(s=Ee.exec(e[l])||[])[1],h=(s[2]||"").split(".").sort(),d&&(f=k.e
o,a,s,u,l,c,f,p,d,h,g,v=Q.hasData(e)&&Q.get(e);if(v&&(u=v.events)){l=(t=(t||"").match(R)||[""]).length;while(l--)if(d=
RegExp("(^|\\.)"+h.join("\\.(?:.*\\.|)")+"(\\.|$)"),a=o=p.length;while(o--)c=p[o],!i&&g!==c.origType||n&&n.guid!==c.gui
u[d])}else for(d in u)k.event.remove(e,d+t[l],n,r,!0);k.isEmptyObject(u)&&Q.remove(e,"handle
events")}},dispatch:function(e){var t,n,r,i,o,a,s=k.event.fix(e),u=new
Array(arguments.length),l=(Q.get(this,"events")||{})[s.type]||[],c=k.event.special[s.type]||{};for(u[0]=s,t=1;t<argume
0!==(r=((k.event.special[o.origType]||{}).handle||o.handler).apply(i.elem,u))&&!1===(s.result=r)&&(s.preventDefault(),
```

```
c.postDispatch&&c.postDispatch.call(this,s),s.result}},handlers:function(e,t){var
n,r,i,o,a,s=[],u=t.delegateCount,l=e.target;if(u&&l.nodeType&&!("click"===e.type&&1<=e.button))for(;l!==this;l=l.parent
0===a[i=(r=t[n]).selector+"
"]&&(a[i]=r.needsContext?-1<k(i,this).index(l):k.find(i,this,null,[l]).length),a[i]&&o.push(r);o.length&&s.push({elem:l
l=this,u<t.length&&s.push({elem:l,handlers:t.slice(u)}),s},addProp:function(t,e){Object.defineProperty(k.Event.prototyp
e(this.originalEvent)}:function(){if(this.originalEvent)return
this.originalEvent[t]},set:function(e){Object.defineProperty(this,t,{enumerable:!0,configurable:!0,writable:!0,value:e
e[k.expando]?e:new k.Event(e)},special:{load:{noBubble:!0},click:{setup:function(e){var
t=this||e;return pe.test(t.type)&&t.click&&A(t,"input")&&De(t,"click",ke),!1},trigger:function(e){var
t=this||e;return pe.test(t.type)&&t.click&&A(t,"input")&&De(t,"click"),!0},_default:function(e){var
t=e.target;return
pe.test(t.type)&&t.click&&A(t,"input")&&Q.get(t,"click")||A(t,"a")}},beforeunload:{postDispatch:function(e){void
0!==e.result&&e.originalEvent&&(e.originalEvent.returnValue=e.result)}}}},k.removeEvent=function(e,t,n){e.removeEventL
instanceof k.Event))return new
k.Event(e,t);e&&e.type?(this.originalEvent=e,this.type=e.type,this.isDefaultPrevented=e.defaultPrevented||void
0===e.defaultPrevented&&!1===e.returnValue?ke:Se,this.target=e.target&&3===e.target.nodeType?e.target.parentNode:e.tar
e=this.originalEvent;this.isDefaultPrevented=ke,e&&!this.isSimulated&&e.preventDefault()},stopPropagation:function(){va
e=this.originalEvent;this.isPropagationStopped=ke,e&&!this.isSimulated&&e.stopPropagation()},stopImmediatePropagation:
e=this.originalEvent;this.isImmediatePropagationStopped=ke,e&&!this.isSimulated&&e.stopImmediatePropagation(),this.stop
t=e.button;return null==e.which&&Te.test(e.type)?null!=e.charCode?e.charCode:e.keyCode:!e.which&&void
0!==t&&Ce.test(e.type)?1&t?1:2&t?3:4&t?2:0:e.which}},k.event.addProp),k.each({focus:"focusin",blur:"focusout"},function
De(this,e,Ne),!1},trigger:function(){return
De(this,e),!0},delegateType:t}}),k.each({mouseenter:"mouseover",mouseleave:"mouseout",pointerenter:"pointerover",pointe
t,n=e.relatedTarget,r=e.handleObj;return
n&&(n===this||k.contains(this,n))||(e.type=r.origType,t=r.handler.apply(this,arguments),e.type=i),t}}}),k.fn.extend({on
Ae(this,e,t,n,r)},one:function(e,t,n,r){return Ae(this,e,t,n,r,1)},off:function(e,t,n){var
r,i;if(e&&e.preventDefault&&e.handleObj)return
r=e.handleObj,k(e.delegateTarget).off(r.namespace?r.origType+"."+r.namespace:r.origType,r.selector,r.handler),this;if("
e){for(i in e)this.off(i,t,e[i]);return this}return!1!==t&&"function"!=typeof t||(n=t,t=void
0),!1===n&&(n=Se),this.each(function(){k.event.remove(this,e,n,t)})}}});var
je=/<(?!area|br|col|embed|hr|img|input|link|meta|param)(([a-z][^\/\0>\x20\t\r\n\f]*)[^>]*)\/>/gi,qe=/<script|<style|<li
Oe(e,t){return
A(e,"table")&&A(11!==t.nodeType?t:t.firstChild,"tr")&&k(e).children("tbody")[0]||e}function
Pe(e){return e.type=(null!==e.getAttribute("type"))+"/"+e.type,e}function
Re(e){return"true/"===(e.type||"").slice(0,5)?e.type=e.type.slice(5):e.removeAttribute("type"),e}function
Me(e,t){var
n,r,i,o,a,s,u,l;if(1===t.nodeType){if(Q.hasData(e)&&(o=Q.access(e),a=Q.set(t,o),l=o.events))for(i in
delete
a.handle,a.events={},l)for(n=0,r=l[i].length;n<r;n++)k.event.add(t,i,l[i][n]);J.hasData(e)&&(s=J.access(e),u=k.extend(
Ie(n,r,i,o){r=g.apply([],r);var
e,t,a,s,u,l,c=0,f=n.length,p=f-1,d=r[0],h=m(d);if(h||1<f&&"string"==typeof
d&&!y.checkClone&&Le.test(d))return n.each(function(e){var
t=n.eq(e);h&&(r[0]=d.call(this,e,t.html())),Ie(t,r,i,o)});if(f&&(t=(e=we(r,n[0].ownerDocument,!1,n,o)).firstChild,1===e
n}function We(e,t,n){for(var
r,i=t?k.filter(t,e):e,o=0;null!=(r=i[o]);o++)n||1!==r.nodeType||k.cleanData(ve(r)),r.parentNode&&(n&&oe(r)&&ye(ve(r,"sc
e}k.extend({htmlPrefilter:function(e){return e.replace(je,"<$1></$2>")},clone:function(e,t,n){var
r,i,o,a,s,u,l,c=e.cloneNode(!0),f=oe(e);if(!(y.noCloneChecked||1!==e.nodeType&&11!==e.nodeType||k.isXMLDoc(e)))for(a=ve
0,"input"===(l=u.nodeName.toLowerCase())&&pe.test(s.type)?u.checked=s.checked:"input"!==l&&"textarea"!==l||(u.defaultVa
Me(e,c);return 0<(a=ve(c,"script")).length&&ye(a,!f&&ve(e,"script")),c},cleanData:function(e){for(var
t,n,r,i=k.event.special,o=0;void 0!==(n=e[o]);o++)if(G(n)){if(t=n[Q.expando]){if(t.events)for(r in
t.events)i[r]?k.event.remove(n,r):k.removeEvent(n,r,t.handle);n[Q.expando]=void
0}n[J.expando]&&(n[J.expando]=void 0)}}}),k.fn.extend({detach:function(e){return
We(this,e,!0)},remove:function(e){return We(this,e)},text:function(e){return
_(this,function(e){return void
0===e?k.text(this):this.empty().each(function(){1!==this.nodeType&&11!==this.nodeType&&9!==this.nodeType||(this.textCon
Ie(this,arguments,function(e){1!==this.nodeType&&11!==this.nodeType&&9!==this.nodeType||Oe(this,e).appendChild(e)})},pr
Ie(this,arguments,function(e){if(1===this.nodeType||11===this.nodeType||9===this.nodeType){var
t=Oe(this,e);t.insertBefore(e,t.firstChild)}})},before:function(){return
Ie(this,arguments,function(e){this.parentNode&&this.parentNode.insertBefore(e,this)})},after:function(){return
Ie(this,arguments,function(e){this.parentNode&&this.parentNode.insertBefore(e,this.nextSibling)})},empty:function(){for
e,t=0;null!=(e=this[t]);t++)1===e.nodeType&&(k.cleanData(ve(e,!1)),e.textContent="");return
this},clone:function(e,t){return e=null!=e&&e,t=null==t?e:t,this.map(function(){return
k.clone(this,e,t)})},html:function(e){return _(this,function(e){var
t=this[0]||{},n=0,r=this.length;if(void 0===e&&1===t.nodeType)return t.innerHTML;if("string"==typeof
e&&!qe.test(e)&&!ge[(de.exec(e)||["",""])[1].toLowerCase()]){e=k.htmlPrefilter(e);try{for(;n<r;n++)1===(t=this[n]||{})
n=[];return Ie(this,arguments,function(e){var
t=this.parentNode;k.inArray(this,n)<0&&(k.cleanData(ve(this)),t&&t.replaceChild(e,this))},n)}}),k.each({appendTo:"appen
t,n=[],r=k(e),i=r.length-1,o=0;o<=i;o++)t=o===i?this:this.clone(!0),k(r[o])[a](t),u.apply(n,t.get());return
this.pushStack(n)}});var $e=new RegExp("^("+te+")(?!px)[a-z%]+$","i"),Fe=function(e){var
t=e.ownerDocument.defaultView;return t&&t.opener||(t=C),t.getComputedStyle(e)},Be=new
RegExp(re.join("|"),"i");function _e(e,t,n){var
r,i,o,a,s=e.style;return(n=n||Fe(e))&&(""!==(a=n.getPropertyValue(t)||n[t])||oe(e)||(a=k.style(e,t)),!y.pixelBoxStyles
0!==a?a+"":a}function
ze(e,t){return{get:function(){if(!e())return(this.get=t).apply(this,arguments);delete
this.get}}}!function(){function
e(){if(u){s.style.cssText="position:absolute;left:-11111px;width:60px;margin-top:1px;padding:0;border:0",u.style.cssTe
e=C.getComputedStyle(u);n="1%"!==e.top,a=12===t(e.marginLeft),u.style.right="60%",o=36===t(e.right),r=36===t(e.width),u
t(e){return Math.round(parseFloat(e))}var
n,r,i,o,a,s=E.createElement("div"),u=E.createElement("div");u.style&&(u.style.backgroundClip="content-box",u.cloneNode
e(),r},pixelBoxStyles:function(){return e(),o},pixelPosition:function(){return
e(),n},reliableMarginLeft:function(){return e(),a},scrollboxSize:function(){return e(),i}}))}();var
Ue=["Webkit","Moz","ms"],Xe=E.createElement("div").style,Ve={};function Ge(e){var
t=k.cssProps[e]||Ve[e];return t||(e in Xe?e:Ve[e]=function(e){var
t=e[0].toUpperCase()+e.slice(1),n=Ue.length;while(n--)if((e=Ue[n]+t)in Xe)return e}(e)||e)}var
Ye=/^(none|table(?!-c[ea]).+)/,Qe=/^--/,Je={position:"absolute",visibility:"hidden",display:"block"},Ke={letterSpacing
Ze(e,t,n){var r=ne.exec(t);return r?Math.max(0,r[2]-(n||0))+(r[3]||"px"):t}function
et(e,t,n,r,i,o){var a="width"===t?1:0,s=0,u=0;if(n===(r?"border":"content"))return
```

```
0;for(;a<4;a+=2)"margin"===n&&(u+=k.css(e,n+re[a],!0,i)),r?("content"===n&&(u-=k.css(e,"padding"+re[a],!0,i)),"margin"
tt(e,t,n){var
r=Fe(e),i=(!y.boxSizingReliable()||n)&&"border-box"===k.css(e,"boxSizing",!1,r),o=i,a=_e(e,t,r),s="offset"+t[0].toUppe
a;a="auto"}return(!y.boxSizingReliable()&&i||"auto"===a||!parseFloat(a)&&"inline"===k.css(e,"display",!1,r))&&e.getClie
in e)&&(a=e[s])),(a=parseFloat(a)||0)+et(e,t,n||(i?"border":"content"),o,r,a)+"px"}function
nt(e,t,n,r,i){return new
nt.prototype.init(e,t,n,r,i)}k.extend({cssHooks:{opacity:{get:function(e,t){if(t){var
n=_e(e,"opacity");return""===n?"1":n}}}},cssNumber:{animationIterationCount:!0,columnCount:!0,fillOpacity:!0,flexGrow:
i,o,a,s=V(t),u=Qe.test(t),l=e.style;if(u||(t=Ge(s)),a=k.cssHooks[t]||k.cssHooks[s],void 0===n)return
a&&"get"in a&&void 0!==(i=a.get(e,!1,r))?i:l[t];"string"===(o=typeof
n)&&(i=ne.exec(n))&&i[1]&&(n=le(e,t,i),o="number"),null!=n&&n==n&&("number"!==o||u||(n+=i&&i[3]||(k.cssNumber[s]?"":"p
a&&void 0===(n=a.set(e,n,r))||(u?l.setProperty(t,n):l[t]=n))}},css:function(e,t,n,r){var
i,o,a,s=V(t);return Qe.test(t)||(t=Ge(s)),(a=k.cssHooks[t]||k.cssHooks[s])&&"get"in
a&&(i=a.get(e,!0,n)),void 0===i&&(i=_e(e,t,r)),"normal"===i&&t in
Ke&&(i=Ke[t]),""===n||n?(o=parseFloat(i),!0===n||isFinite(o)?o||0:i):i}}),k.each(["height","width"],function(e,u){k.css
tt(e,u,n)})},set:function(e,t,n){var
r,i=Fe(e),o=!y.scrollboxSize()&&"absolute"===i.position,a=(o||n)&&"border-box"===k.css(e,"boxSizing",!1,i),s=n?et(e,u,
a&&o&&(s-=Math.ceil(e["offset"+u[0].toUpperCase()+u.slice(1)]-parseFloat(i[u])-et(e,u,"border",!1,i)-.5)),s&&(r=ne.exec
e.getBoundingClientRect().left}))+"px"},k.each({margin:"",padding:"",border:"Width"},function(i,o){k.cssHooks[i+o]={ex
t=0,n={},r="string"==typeof e?e.split(" "):[e];t<4;t++)n[i+re[t]+o]=r[t]||r[t-2]||r[0];return
n}},"margin"!==i&&(k.cssHooks[i+o].set=Ze)},k.fn.extend({css:function(e,t){return
_(this,function(e,t,n){var
r,i,o={},a=0;if(Array.isArray(t)){for(r=Fe(e),i=t.length;a<i;a++)o[t[a]]=k.css(e,t[a],!1,r);return
o}return void
0!==n?k.style(e,t,n):k.css(e,t)},e,t,l<arguments.length)}}),((k.Tween=nt).prototype={constructor:nt,init:function(e,t,
e=nt.propHooks[this.prop];return
e&&e.get?e.get(this):nt.propHooks._default.get(this)},run:function(e){var
t,n=nt.propHooks[this.prop];return
this.options.duration?this.pos=t=k.easing[this.easing](e,this.options.duration*e,0,1,this.options.duration):this.pos=t=
t;return
l!==e.elem.nodeType||null!=e.elem[e.prop]&&null==e.elem.style[e.prop]?e.elem[e.prop]:(t=k.css(e.elem,e.prop,""))&&"auto
e},swing:function(e){return.5-Math.cos(e*Math.PI)/2},_default:"swing"},k.fx=nt.prototype.init,k.fx.step={};var
rt,it,ot,at,st=/^(?:toggle|show|hide)$/,ut=/queueHooks$/;function
lt(){it&&(!l===E.hidden&&C.requestAnimationFrame?C.requestAnimationFrame(lt):C.setTimeout(lt,k.fx.interval),k.fx.tick()
ct(){return C.setTimeout(function(){rt=void 0}),rt=Date.now()}function ft(e,t){var
n,r=0,i={height:e};for(t=t?1:0;r<4;r+=2-t)i["margin"+(n=re[r])]=i["padding"+n]=e;return
t&&(i.opacity=i.width=e),i}function pt(e,t,n){for(var
r,i=(dt.tweeners[t]||[]).concat(dt.tweeners["*"]),o=0,a=i.length;o<a;o++)if(r=i[o].call(n,t,e))return
r}function dt(o,e,t){var n,a,r=0,i=dt.prefilters.length,s=k.Deferred().always(function(){delete
u.elem}),u=function(){if(a)return!1;for(var
e=rt||ct(),t=Math.max(0,l.startTime+l.duration-e),n=1-(t/l.duration||0),r=0,i=l.tweens.length;r<i;r++)l.tweens[r].run(n
s.notifyWith(o,[l,n,t]),n<1&&i?t:(i||s.notifyWith(o,[l,1,0]),s.resolveWith(o,[l]),!1)},l=s.promise({elem:o,props:k.exte
n=k.Tween(o,l.opts,e,t,l.opts.specialEasing[e]||l.opts.easing);return
l.tweens.push(n),n},stop:function(e){var t=0,n=e?l.tweens.length:0;if(a)return
this;for(a=!0;t<n;t++)l.tweens[t].run(1);return
e?(s.notifyWith(o,[l,1,0]),s.resolveWith(o,[l,e])):s.rejectWith(o,[l,e]),this}}),c=l.props;for(!function(e,t){var
n,r,i,o,a;for(n in
e)if(i=t[r=V(n)],o=e[n],Array.isArray(o)&&(i=o[1],o=e[n]=o[0]),n!==r&&(e[r]=o,delete
e[n]),(a=k.cssHooks[r])&&"expand"in a)for(n in o=a.expand(o),delete e[r],o)n in
e||(e[n]=o[n],t[n]=i);else
t[r]=i}(c,l.opts.specialEasing);r<i;r++)if(n=dt.prefilters[r].call(l,o,c,l.opts))return
m(n.stop)&&(k._queueHooks(l.elem,l.opts.queue).stop=n.stop.bind(n)),n;return
k.map(c,pt,l),m(l.opts.start)&&l.opts.start.call(o,l),l.progress(l.opts.progress).done(l.opts.done,l.opts.complete).fai
n=this.createTween(e,t);return
le(n.elem,e,ne.exec(t),n),n}]},tweener:function(e,t){m(e)?(t=e,e=["*"]):e=e.match(R);for(var
n,r=0,i=e.length;r<i;r++)n=e[r],dt.tweeners[n]=dt.tweeners[n]||[],dt.tweeners[n].unshift(t)},prefilters:[function(e,t,
r,i,o,a,s,u,l,c,f="width"in t||"height"in
t,p=this,d={},h=e.style,g=e.nodeType&&se(e),v=Q.get(e,"fxshow");for(r in
n.queue||(null==(a=k._queueHooks(e,"fx")).unqueued&&(a.unqueued=0,s=a.empty.fire,a.empty.fire=function(){a.unqueued||s
t[r],o=o||"toggle"===i,i===(g?"hide":"show")){if("show"!==i||!v||void
0===v[r])continue;g=!0}d[r]=v&&v[r]||k.style(e,r)}if((u=!k.isEmptyObject(t))||!k.isEmptyObject(d))for(r
in
f&&1===e.nodeType&&(n.overflow=[h.overflow,h.overflowX,h.overflowY],null==(l=v&&v.display)&&(l=Q.get(e,"display")),"no
v&&(g=v.hidden):v=Q.access(e,"fxshow",{display:l}),o&&(v.hidden=!g),g&&fe([e],!0),p.done(function(){for(r
in g||fe([e]),Q.remove(e,"fxshow"),d)k.style(e,r,d[r])})),u=pt(g?v[r]:0,r,p),r in
v||(v[r]=u.start,g&&(u.end=u.start,u.start=0))}],prefilter:function(e,t){t?dt.prefilters.unshift(e):dt.prefilters.push
r=e&&"object"==typeof
e?k.extend({},e):{complete:n||!n&&t||m(e)&&e,duration:e,easing:n&&t||t&&!m(t)&&t};return
k.fx.off?r.duration=0:"number"!=typeof r.duration&&(r.duration in
k.fx.speeds?r.duration=k.fx.speeds[r.duration]:r.duration=k.fx.speeds._default),null!=r.queue&&!0!==r.queue||(r.queue=
this.filter(se).css("opacity",0).show().end().animate({opacity:t},e,n,r)},animate:function(t,e,n,r){var
i=k.isEmptyObject(t),o=k.speed(e,n,r),a=function(){var
e=dt(this,k.extend({},t),o);(i||Q.get(this,"finish"))&&e.stop(!0)};return
a.finish=a,i||!1===o.queue?this.each(a):this.queue(o.queue,a)},stop:function(i,e,o){var
a=function(e){var t=e.stop;delete e.stop,t(o)};return"string"!=typeof i&&(o=e,e=i,i=void
0),e&&!1!==i&&this.queue(i||"fx",[]),this.each(function(){var
e=!0,t=null!=i&&i+"queueHooks",n=k.timers,r=Q.get(this);if(t)r[t]&&r[t].stop&&a(r[t]);else for(t in
r)r[t]&&r[t].stop&&ut.test(t)&&a(r[t]);for(t=n.length;t--;)n[t].elem!==this||null!=i&&n[t].queue!==i||(n[t].anim.stop(
e,t=Q.get(this),n=t[a+"queue"],r=t[a+"queueHooks"],i=k.timers,o=n?n.length:0;for(t.finish=!0,k.queue(this,a,[]),r&&r.st
t.finish})}}),k.each(["toggle","show","hide"],function(e,r){var
i=k.fn[r];k.fn[r]=function(e,t,n){return null==e||"boolean"==typeof
e?i.apply(this,arguments):this.animate(ft(r,!0),e,t,n)}}),k.each({slideDown:ft("show"),slideUp:ft("hide"),slideToggle:
this.animate(r,e,t,n)}}),k.timers=[],k.fx.tick=function(){var
e,t=0,n=k.timers;for(rt=Date.now();t<n.length;t++)(e=n[t])()||n[t]!==e||n.splice(t--,1);n.length||k.fx.stop(),rt=void
0},k.fx.timer=function(e){k.timers.push(e),k.fx.start()},k.fx.interval=13,k.fx.start=function(){it||(it=!0,lt())},k.fx
r=k.fx&&k.fx.speeds[r]||r,e=e||"fx",this.queue(e,function(e,t){var
n=C.setTimeout(e,r);t.stop=function(){C.clearTimeout(n)}})},ot=E.createElement("input"),at=E.createElement("select").ap
```

```
ht,gt=k.expr.attrHandle;k.fn.extend({attr:function(e,t){return
_(this,k.attr,e,t,1<arguments.length)},removeAttr:function(e){return
this.each(function(){k.removeAttr(this,e)})}}),k.extend({attr:function(e,t,n){var
r,i,o=e.nodeType;if(3!==o&&8!==o&&2!==o)return"undefined"==typeof
e.getAttribute?k.prop(e,t,n):(1===o&&k.isXMLDoc(e)||(i=k.attrHooks[t.toLowerCase()]||(k.expr.match.bool.test(t)?ht:voi
0)),void 0!==n?null===n?void k.removeAttr(e,t):i&&"set"in i&&void
0!==(r=i.set(e,n,t))?r:(e.setAttribute(t,n+""),n):i&&"get"in
i&&null!==(r=i.get(e,t))?r:null==(r=k.find.attr(e,t))?void
0:r)},attrHooks:{type:{set:function(e,t){if(!y.radioValue&&"radio"===t&&A(e,"input")){var
n=e.value;return e.setAttribute("type",t),n&&(e.value=n),t}}}},removeAttr:function(e,t){var
n,r=0,i=t&&t.match(R);if(i&&1===e.nodeType)while(n=i[r++])e.removeAttribute(n)}}),ht={set:function(e,t,n){return!1===t
a=gt[t]||k.find.attr;gt[t]=function(e,t,n){var r,i,o=t.toLowerCase();return
n||(i=gt[o],gt[o]=r,r=null!=a(e,t,n)?o:null,gt[o]=i),r}});var
vt=/^(?:input|select|textarea|button)$/i,yt=/^(?:a|area)$/i;function
mt(e){return(e.match(R)||[]).join(" ")}function xt(e){return
e.getAttribute&&e.getAttribute("class")||""}function bt(e){return Array.isArray(e)?e:"string"==typeof
e&&e.match(R)||[]}k.fn.extend({prop:function(e,t){return
_(this,k.prop,e,t,1<arguments.length)},removeProp:function(e){return this.each(function(){delete
this[k.propFix[e]||e]})}}),k.extend({prop:function(e,t,n){var
r,i,o=e.nodeType;if(3!==o&&8!==o&&2!==o)return
1===o&&k.isXMLDoc(e)||(t=k.propFix[t]||t,i=k.propHooks[t]),void 0!==n?i&&"set"in i&&void
0!==(r=i.set(e,n,t))?r:e[t]=n:i&&"get"in
i&&null!==(r=i.get(e,t))?r:e[t]},propHooks:{tabIndex:{get:function(e){var
t=k.find.attr(e,"tabindex");return
t?parseInt(t,10):vt.test(e.nodeName)||yt.test(e.nodeName)&&e.href?0:-1}}},propFix:{"for":"htmlFor","class":"className"
t=e.parentNode;return t&&t.parentNode&&t.parentNode.selectedIndex,null},set:function(e){var
t=e.parentNode;t&&(t.selectedIndex,t.parentNode&&t.parentNode.selectedIndex)}}),k.each(["tabIndex","readOnly","maxLengt
e,n,r,i,o,a,s,u=0;if(m(t))return
this.each(function(e){k(this).addClass(t.call(this,e,xt(this)))});if((e=bt(t)).length)while(n=this[u++])if(i=xt(n),r=1=
"+mt(i)+" "){a=0;while(o=e[a++])r.indexOf(" "+o+" ")<0&&(r+=o+"
");i!==(s=mt(r))&&n.setAttribute("class",s)}return this},removeClass:function(t){var
e,n,r,i,o,a,s,u=0;if(m(t))return
this.each(function(e){k(this).removeClass(t.call(this,e,xt(this)))});if(!arguments.length)return
this.attr("class","");if((e=bt(t)).length)while(n=this[u++])if(i=xt(n),r=1===n.nodeType&&" "+mt(i)+"
"){a=0;while(o=e[a++])while(-1<r.indexOf(" "+o+" "))r=r.replace(" "+o+" ","
");i!==(s=mt(r))&&n.setAttribute("class",s)}return this},toggleClass:function(i,t){var o=typeof
i,a="string"===o||Array.isArray(i);return"boolean"==typeof
t&&a?t?this.addClass(i):this.removeClass(i):m(i)?this.each(function(e){k(this).toggleClass(i.call(this,e,xt(this),t),t)
e,t,n,r;if(a){t=0,n=k(this),r=bt(i);while(e=r[t++])n.hasClass(e)?n.removeClass(e):n.addClass(e)}else
void
0!==i&&"boolean"!==o||((e=xt(this))&&Q.set(this,"__className__",e),this.setAttribute&&this.setAttribute("class",e||!1==
t,n,r=0;t=" "+e+" ";while(n=this[r++])if(1===n.nodeType&&-1<(" "+mt(xt(n))+"
").indexOf(t))return!0;return!1}});var wt=/\r/g;k.fn.extend({val:function(n){var
r,e,i,t=this[0];return arguments.length?(i=m(n),this.each(function(e){var
t;1===this.nodeType&&(null==(t=i?n.call(this,e,k(this).val()):n)?t="":"number"==typeof
t?t+="":Array.isArray(t)&&(t=k.map(t,function(e){return
null==e?"":e+""})),(r=k.valHooks[this.type]||k.valHooks[this.nodeName.toLowerCase()])&&"set"in
r&&void
0!==r.set(this,t,"value")||(this.value=t))})):t?(r=k.valHooks[t.type]||k.valHooks[t.nodeName.toLowerCase()])&&"get"in
r&&void 0!==(e=r.get(t,"value"))?e:"string"==typeof(e=t.value)?e.replace(wt,""):null==e?"":e:void
0}}),k.extend({valHooks:{option:{get:function(e){var t=k.find.attr(e,"value");return
null!=t?t:mt(k.text(e))}},select:{get:function(e){var
t,n,r,i=e.options,o=e.selectedIndex,a="select-one"===e.type,s=a?null:[],u=a?o+1:i.length;for(r=o<0?u:a?o:0;r<u;r++)if(
t;s.push(t)}return s},set:function(e,t){var
n,r,i=e.options,o=k.makeArray(t),a=i.length;while(a--)((r=i[a]).selected=-1<k.inArray(k.valHooks.option.get(r),o))&&(n=
n||(e.selectedIndex=-1),o}}}}),k.each(["radio","checkbox"],function(){k.valHooks[this]={set:function(e,t){if(Array.isA
e.checked=-1<k.inArray(k(e).val(),t)}},y.checkOn||(k.valHooks[this].get=function(e){return
null===e.getAttribute("value")?"on":e.value})}),y.focusin="onfocusin"in C;var
Tt=/^(?:focusinfocus|focusoutblur)$/,Ct=function(e){e.stopPropagation()};k.extend(k.event,{trigger:function(e,t,n,r){va
i,o,a,s,u,l,c,f,p=[n||E],d=v.call(e,"type")?e.type:e,h=v.call(e,"namespace")?e.namespace.split("."):[];if(o=f=a=n=n||E,
k.Event(d,"object"==typeof
e&&e)).isTrigger=r?2:3,e.namespace=h.join(".",e.rnamespace=e.namespace?new
RegExp("(^|\\.)"+h.join("\\.(?:.*\\.|)")+"(\\.|$)"):null,e.result=void
0,e.target||(e.target=n),t=null==t?[e]:k.makeArray(t,[e]),c=k.event.special[d]||{},r||!c.trigger||!1!==c.trigger.apply
e.type=d,r||e.isDefaultPrevented()||c._default&&!1!==c._default.apply(p.pop(),t)||!G(n)||u&&m(n[d])&&!x(n)&&((a=n[u])&&
0,a&&(n[u]=a)),e.result}},simulate:function(e,t,n){var r=k.extend(new
k.Event,n,{type:e,isSimulated:!0});k.event.trigger(r,null,t)}}),k.fn.extend({trigger:function(e,t){return
this.each(function(){k.event.trigger(e,t,this)})},triggerHandler:function(e,t){var
n=this[0];if(n)return
k.event.trigger(e,t,n,!0)}}),y.focusin||k.each({focus:"focusin",blur:"focusout"},function(n,r){var
i=function(e){k.event.simulate(r,e.target,k.event.fix(e))};k.event.special[r]={setup:function(){var
e=this.ownerDocument||this,t=Q.access(e,r);t||e.addEventListener(n,i,!0),Q.access(e,r,(t||0)+1)},teardown:function(){va
e=this.ownerDocument||this,t=Q.access(e,r)-1;t?Q.access(e,r,t):(e.removeEventListener(n,i,!0),Q.remove(e,r))}}});var
Et=C.location,kt=Date.now(),St=/\?/;k.parseXML=function(e){var t;if(!e||"string"!=typeof e)return
null;try{t=(new C.DOMParser).parseFromString(e,"text/xml")}catch(e){t=void 0}return
t&&!t.getElementsByTagName("parsererror").length||k.error("Invalid XML: "+e),t};var
Nt=/\[\]$/,At=/\r?\n/g,Dt=/^(?:submit|button|image|reset|file)$/i,jt=/^(?:input|select|textarea|keygen)/i;function
qt(n,e,r,i){var
t;if(Array.isArray(e))k.each(e,function(e,t){r||Nt.test(n)?i(n,t):qt(n+"["+("object"==typeof
t&&null!=t?e:"")+"]",t,r,i)});else if(r||"object"!==w(e))i(n,e);else for(t in
e)qt(n+"["+t+"]",e[t],r,i)}k.param=function(e,t){var n,r=[],i=function(e,t){var
n=m(t)?t():t;r[r.length]=encodeURIComponent(e)+"="+encodeURIComponent(null==n?"":n)};if(null==e)return"";if(Array.isAr
for(n in e)qt(n,e[n],t,i);return r.join("&")},k.fn.extend({serialize:function(){return
k.param(this.serializeArray())},serializeArray:function(){return this.map(function(){var
e=k.prop(this,"elements");return e?k.makeArray(e):this}).filter(function(){var e=this.type;return
this.name&&!k(this).is(":disabled")&&jt.test(this.nodeName)&&!Dt.test(e)&&(this.checked||!pe.test(e))}).map(function(e,
n=k(this).val();return
```

```
null==n?null:Array.isArray(n)?k.map(n,function(e){return{name:t.name,value:e.replace(At,"\r\n")}}):{name:t.name,value:n
Lt=/%20/g,Ht=/#.*$/,Ot=/([?&])_=[^&]*/,Pt=/^(.*?):[
\t]*([^\r\n]*)$/gm,Rt=/^(?:GET|HEAD)$/,Mt=/^\/\//,It={},Wt={},$t="*/".concat("*"),Ft=E.createElement("a");function
Bt(o){return function(e,t){"string"!=typeof e&&(t=e,e="*");var
n,r=0,i=e.toLowerCase().match(R)||[];if(m(t))while(n=i[r++])"+"===n[0]?(n=n.slice(1)||"*",(o[n]=o[n]||[]).unshift(t)):
_t(t,i,o,a){var s={},u=t===Wt;function l(e){var r;return s[e]=!0,k.each(t[e]||[],function(e,t){var
n=t(i,o,a);return"string"!=typeof n||u||s[n]?u?!(r=n):void
0:(i.dataTypes.unshift(n),l(n),!1)}),r}return l(i.dataTypes[0])||!s["*"]&&l("*")}function zt(e,t){var
n,r,i=k.ajaxSettings.flatOptions||{};for(n in t)void 0!==t[n]&&((i[n]?e:r||(r={}))[n]=t[n]);return
r&&k.extend(!0,e,r),e}Ft.href=Et.href,k.extend({active:0,lastModified:{},etag:{},ajaxSettings:{url:Et.href,type:"GET",i
charset=UTF-8",accepts:{"*":$t,text:"text/plain",html:"text/html",xml:"application/xml,
text/xml",json:"application/json,
text/javascript"},contents:{xml:/\bxml\b/,html:/\bhtml/,json:/\bjson\b/},responseFields:{xml:"responseXML",text:"respon
text":String,"text html":!0,"text json":JSON.parse,"text
xml":k.parseXML},flatOptions:{url:!0,context:!0}},ajaxSetup:function(e,t){return
t?zt(zt(e,k.ajaxSettings),t):zt(k.ajaxSettings,e)},ajaxPrefilter:Bt(It),ajaxTransport:Bt(Wt),ajax:function(e,t){"object
e&&(t=e,e=void 0),t=t||{};var
c,f,p,n,d,r,h,g,i,o,v=k.ajaxSetup({},t),y=v.context||v,m=v.context&&(y.nodeType||y.jquery)?k(y):k.event,x=k.Deferred(),
memory"),w=v.statusCode||{},a={},s={},u="canceled",T={readyState:0,getResponseHeader:function(e){var
t;if(h){if(!n){n={};while(t=Pt.exec(p))n[t[1].toLowerCase()+" "]=(n[t[1].toLowerCase()+"
"]||[]).concat(t[2])}t=n[e.toLowerCase()+" "]}return null==t?null:t.join(",
")},getAllResponseHeaders:function(){return h?p:null},setRequestHeader:function(e,t){return
null==h&&(e=s[e.toLowerCase()]=s[e.toLowerCase()]||e,a[e]=t),this},overrideMimeType:function(e){return
null==h&&(v.mimeType=e),this},statusCode:function(e){var t;if(e)if(h)T.always(e[T.status]);else for(t
in e)w[t]=[w[t],e[t]];return this},abort:function(e){var t=e||u;return
c&&c.abort(t),l(0,t),this}};if(x.promise(T),v.url=((e||v.url||Et.href)+"").replace(Mt,Et.protocol+"//"),v.type=t.method
v.data&&(v.data=k.param(v.data,v.traditional)),_t(It,v,t,T),h)return T;for(i
in(g=k.event&&v.global)&&0==k.active++&&k.event.trigger("ajaxStart"),v.type=v.type.toUpperCase(),v.hasContent=!Rt.test
v.data)&&(f+=(St.test(f)?"&":"?")+v.data,delete
v.data),!1===v.cache&&(f=f.replace(Ot,"$1"),o=(St.test(f)?"&":"?")+"_="+kt+++o),v.url=f+o),v.ifModified&&(k.lastModifie
"+$t+";
q=0.01":""):v.accepts["*"]),v.headers)T.setRequestHeader(i,v.headers[i]);if(v.beforeSend&&(!1===v.beforeSend.call(y,T,v
T.abort();if(u="abort",b.add(v.complete),T.done(v.success),T.fail(v.error),c=_t(Wt,v,t,T)){if(T.readyState=1,g&&m.trigg
T;v.async&&0<v.timeout&&(d=C.setTimeout(function(){T.abort("timeout")},v.timeout));try{h=!1,c.send(a,l)}catch(e){if(h)t
e;l(-1,e)}}else l(-1,"No Transport");function l(e,t,n,r){var
i,o,a,s,u,l=t;h||(h=!0,d&&C.clearTimeout(d),c=void
0,p=r||"",T.readyState=0<e?4:0,i=200<=e&&e<300||304===e,n&&(s=function(e,t,n){var
r,i,o,a,s=e.contents,u=e.dataTypes;while("*"===u[0])u.shift(),void
0===r&&(r=e.mimeType||t.getResponseHeader("Content-Type"));if(r)for(i in
s)if(s[i]&&s[i].test(r)){u.unshift(i);break}if(u[0]in n)o=u[0];else{for(i in
n){if(!u[0]||e.converters[i+" "+u[0]]){o=i;break}a||(a=i)}o=o||a}if(o)return
o!==u[0]&&u.unshift(o),n[o]}(v,T,n)),s=function(e,t,n,r){var
i,o,a,s,u,l={},c=e.dataTypes.slice();if(c[1])for(a in
e.converters)l[a.toLowerCase()]=e.converters[a];o=c.shift();while(o)if(e.responseFields[o]&&(n[e.responseFields[o]]=t),
if("*"!==u&&u!==o){if(!(a=l[u+" "+o]||l["* "+o]))for(i in l)if((s=i.split(" "))[1]===o&&(a=l[u+"
"+s[0]]||l["*
"+s[0]]){!0===a?a=l[i]:!0!==l[i]&&(o=s[0],c.unshift(s[1]));break}if(!0!==a)if(a&&e["throws"])t=a(t);else
try{t=a(t)}catch(e){return{state:"parsererror",error:a?e:"No conversion from "+u+" to
"+o}}}return{state:"success",data:t}}(v,s,T,i),i?(v.ifModified&&((u=T.getResponseHeader("Last-Modified"))&&(k.lastModi
0,g&&m.trigger(i?"ajaxSuccess":"ajaxError",[T,v,i?o:a]),b.fireWith(y,[T,l]),g&&(m.trigger("ajaxComplete",[T,v]),--k.act
T},getJSON:function(e,t,n){return k.get(e,t,n,"json")},getScript:function(e,t){return k.get(e,void
0,t,"script")}}),k.each(["get","post"],function(e,i){k[i]=function(e,t,n,r){return
m(t)&&(r=r||n,n=t,t=void
0),k.ajax(k.extend({url:e,type:i,dataType:r,data:t,success:n},k.isPlainObject(e)&&e))}}),k._evalUrl=function(e,t){retur
k.ajax({url:e,type:"GET",dataType:"script",cache:!0,async:!1,global:!1,converters:{"text
script":function(){}},dataFilter:function(e){k.globalEval(e,t)}})},k.fn.extend({wrapAll:function(e){var
t;return
this[0]&&(m(e)&&(e=e.call(this[0])),t=k(e,this[0].ownerDocument).eq(0).clone(!0),this[0].parentNode&&t.insertBefore(thi
e=this;while(e.firstElementChild)e=e.firstElementChild;return
e}).append(this)),this},wrapInner:function(n){return
m(n)?this.each(function(e){k(this).wrapInner(n.call(this,e))}):this.each(function(){var
e=k(this),t=e.contents();t.length?t.wrapAll(n):e.append(n)})},wrap:function(t){var n=m(t);return
this.each(function(e){k(this).wrapAll(n?t.call(this,e):t)})},unwrap:function(e){return
this.parent(e).not("body").each(function(){k(this).replaceWith(this.childNodes)}),this}}),k.expr.pseudos.hidden=functio
new C.XMLHttpRequest}catch(e){}};var
Ut={0:200,1223:204},Xt=k.ajaxSettings.xhr();y.cors=!!Xt&&"withCredentials"in
Xt,y.ajax=Xt=!!Xt,k.ajaxTransport(function(i){var
o,a;if(y.cors||Xt&&!i.crossDomain)return{send:function(e,t){var
n,r=i.xhr();if(r.open(i.type,i.url,i.async,i.username,i.password),i.xhrFields)for(n in
i.xhrFields)r[n]=i.xhrFields[n];for(n in
i.mimeType&&r.overrideMimeType&&r.overrideMimeType(i.mimeType),i.crossDomain||e["X-Requested-With"]||(e["X-Requested-Wi
function(){o&&(o=a=r.onload=r.onerror=r.onabort=r.ontimeout=r.onreadystatechange=null,"abort"===e?r.abort():"error"===e
r.status?t(0,"error"):t(r.status,r.statusText):t(Ut[r.status]||r.status,r.statusText,"text"!==(r.responseType||"text")
r.responseText?{binary:r.response}:{text:r.responseText},r.getAllResponseHeaders()))}},r.onload=o(),a=r.onerror=r.onti
0!==r.onabort?r.onabort=a:r.onreadystatechange=function(){4===r.readyState&&C.setTimeout(function(){o&&a()})},o=o("abo
e}},abort:function(){o&&o()}}}),k.ajaxPrefilter(function(e){e.crossDomain&&(e.contents.script=!1)}),k.ajaxSetup({accept
application/javascript, application/ecmascript,
application/x-ecmascript"},contents:{script:/\b(?:java|ecma)script\b/},converters:{"text
script":function(e){return k.globalEval(e),e}}}),k.ajaxPrefilter("script",function(e){void
0===e.cache&&(e.cache=!1),e.crossDomain&&(e.type="GET")}),k.ajaxTransport("script",function(n){var
r,i;if(n.crossDomain||n.scriptAttrs)return{send:function(e,t){r=k("<script>").attr(n.scriptAttrs||{}).prop({charset:n.s
error",i=function(e){r.remove(),i=null,e&&t("error"===e.type?404:200,e.type)}),E.head.appendChild(r[0])},abort:functio
Vt,Gt=[],Yt=/(=)\?(?=&|$)|\?\?/;k.ajaxSetup({jsonp:"callback",jsonpCallback:function(){var
e=Gt.pop()||k.expando+"_"+kt++;return this[e]=!0,e}}),k.ajaxPrefilter("json
jsonp",function(e,t,n){var r,i,o,a=!1!==e.jsonp&&(Yt.test(e.url)?"url":"string"==typeof
e.data&&0===(e.contentType||"").indexOf("application/x-www-form-urlencoded")&&Yt.test(e.data)&&"data");if(a||"jsonp"===
r=e.jsonpCallback=m(e.jsonpCallback)?e.jsonpCallback():e.jsonpCallback,a?e[a]=e[a].replace(Yt,"$1"+r):!1!==e.jsonp&&(e
```

```
        json"]=function(){return o||k.error(r+" was not
        called"),o[0]},e.dataTypes[0]="json",i=C[r],C[r]=function(){o=arguments},n.always(function(){void
        0===i?k(C).removeProp(r):C[r]=i,e[r]&&(e.jsonpCallback=t.jsonpCallback,Gt.push(r)),o&&m(i)&&i(o[0]),o=i=void
        0}),"script"}),y.createHTMLDocument=((Vt=E.implementation.createHTMLDocument("").body).innerHTML="<form></form><form><
        e?[]:("boolean"==typeof
        t&&(n=t,t=!1),t||(y.createHTMLDocument?((r=(t=E.implementation.createHTMLDocument("")).createElement("base")).href=E.lo
        r,i,o},k.fn.load=function(e,t,n){var r,i,o,a=this,s=e.indexOf("
        ");return-1<s&&(r=mt(e.slice(s)),e=e.slice(0,s)),m(t)?(n=t,t=void 0):t&&"object"==typeof
        t&&(i="POST"),0<a.length&&k.ajax({url:e,type:i||"GET",dataType:"html",data:t}).done(function(e){o=arguments,a.html(r?k
        this.on(t,e)}}),k.expr.pseudos.animated=function(t){return k.grep(k.timers,function(e){return
        t===e.elem}).length},k.offset={setOffset:function(e,t,n){var
        r,i,o,a,s,u,l=k.css(e,"position"),c=k(e),f={};"static"===l&&(e.style.position="relative"),s=c.offset(),o=k.css(e,"top")
        t?t.using.call(e,f):c.css(f)}},k.fn.extend({offset:function(t){if(arguments.length)return void
        0===t?this:this.each(function(e){k.offset.setOffset(this,t,e)});var e,n,r=this[0];return
        r?r.getClientRects().length?(e=r.getBoundingClientRect(),n=r.ownerDocument.defaultView,{top:e.top+n.pageYOffset,left:e
        0},position:function(){if(this[0]){var
        e,t,n,r=this[0],i={top:0,left:0};if("fixed"===k.css(r,"position"))t=r.getBoundingClientRect();else{t=this.offset(),n=r
        this.map(function(){var
        e=this.offsetParent;while(e&&"static"===k.css(e,"position"))e=e.offsetParent;return
        e||ie})}}),k.each({scrollLeft:"pageXOffset",scrollTop:"pageYOffset"},function(t,i){var
        o="pageYOffset"===i;k.fn[t]=function(e){return _(this,function(e,t,n){var
        r;if(x(e)?r=e:9===e.nodeType&&(r=e.defaultView),void 0===n)return
        r?r[i]:e[t];r?r.scrollTo(o?r.pageXOffset:n,o?n:r.pageYOffset):e[t]=n},t,e,arguments.length)}}),k.each(["top","left"],fu
        t=_e(e,n),$e.test(t)?k(e).position()[n]+"px":t})}),k.each({Height:"height",Width:"width"},function(a,s){k.each({paddin
        n=arguments.length&&(r||"boolean"!=typeof e),i=r||(!0===e||!0===t?"margin":"border");return
        _(this,function(e,t,n){var r;return
        x(e)?0===o.indexOf("outer")?e["inner"+a]:e.document.documentElement["client"+a]:9===e.nodeType?(r=e.documentElement,Mat
        0===n?k.css(e,t,i):k.style(e,t,n,i)},s,n?e:void 0,n)}})}),k.each("blur focus focusin focusout resize
        scroll click dblclick mousedown mouseup mousemove mouseover mouseout mouseenter mouseleave change
        select submit keydown keypress keyup contextmenu".split("
        "),function(e,n){k.fn[n]=function(e,t){return
        0<arguments.length?this.on(n,null,e,t):this.trigger(n)}}),k.fn.extend({hover:function(e,t){return
        this.mouseenter(e).mouseleave(t||e)}}),k.fn.extend({bind:function(e,t,n){return
        this.on(e,null,t,n)},unbind:function(e,t){return
        this.off(e,null,t)},delegate:function(e,t,n,r){return
        this.on(t,e,n,r)},undelegate:function(e,t,n){return
        1===arguments.length?this.off(e,"**"):this.off(t,e||"**",n)}}),k.proxy=function(e,t){var
        n,r,i;if("string"==typeof t&&(n=e[t],t=e,e=n),m(e))return r=s.call(arguments,2),(i=function(){return
        e.apply(t||this,r.concat(s.call(arguments)))}).guid=e.guid=e.guid||k.guid++,i},k.holdReady=function(e){e?k.readyWait++
        t=k.type(e);return("number"===t||"string"===t)&&!isNaN(e-parseFloat(e))},"function"==typeof
        define&&define.amd&&define("jquery",[],function(){return k});var Qt=C.jQuery,Jt=C.$;return
        k.noConflict=function(e){return
        C.$===k&&(C.$=Jt),e&&C.jQuery===k&&(C.jQuery=Qt),k},e||(C.jQuery=C.$=k),k});
/*! jQuery UI - v1.12.1 - 2019-01-27
* http://jqueryui.com
* Includes: widget.js, position.js, data.js, disable-selection.js, focusable.js, form-reset-mixin.js,
        jquery-1-7.js, keycode.js, labels.js, scroll-parent.js, tabbable.js, unique-id.js,
        widgets/resizable.js, widgets/mouse.js
* Copyright jQuery Foundation and other contributors; Licensed MIT */

(function(t){"function"==typeof
        define&&define.amd?define(["jquery"],t):t(jQuery)})(function(t){function e(t){for(var
        e=t.css("visibility");"inherit"===e;)t=t.parent(),e=t.css("visibility");return"hidden"!==e}t.ui=t.ui||{},t.ui.version="
        i=0,s=Array.prototype.slice;t.cleanData=function(e){return function(i){var
        s,n,o;for(o=0;null!=(n=i[o]);o++)try{s=t._data(n,"events"),s&&s.remove&&t(n).triggerHandler("remove")}catch(a){}e(i)}}
        n,o,a,r={},h=e.split(".")[0];e=e.split(".")[1];var l=h+"-"+e;return
        s||(s=i,i=t.Widget),t.isArray(s)&&(s=t.extend.apply(null,[{}].concat(s))),t.expr[":"][l.toLowerCase()]=function(e){retu
        this._createWidget?(arguments.length&&this._createWidget(t,e),void 0):new
        o(t,e)},t.extend(o,n,{version:s.version,_proto:t.extend({},s),_childConstructors:[]}),a=new
        i,a.options=t.widget.extend({},a.options),t.each(s,function(e,s){return
        t.isFunction(s)?(r[e]=function(){function t(){return i.prototype[e].apply(this,arguments)}function
        n(t){return i.prototype[e].apply(this,t)}return function(){var
        e,i=this._super,o=this._superApply;return
        this._super=t,this._superApply=n,e=s.apply(this,arguments),this._super=i,this._superApply=o,e}}(),void
        0):(r[e]=s,void
        0)}),o.prototype=t.widget.extend(a,{widgetEventPrefix:n?a.widgetEventPrefix||e:e},r,{constructor:o,namespace:h,widgetNa
        s=i.prototype;t.widget(s.namespace+"."+s.widgetName,o,i._proto)}),delete
        n._childConstructors):i._childConstructors.push(o),t.widget.bridge(e,o),o},t.widget.extend=function(e){for(var
        i,n,o=s.call(arguments,1),a=0,r=o.length;r>a;a++)for(i in o[a])n=o[a][i],o[a].hasOwnProperty(i)&&void
        0!==n&&(e[i]=t.isPlainObject(n)?t.isPlainObject(e[i])?t.widget.extend({},e[i],n):t.widget.extend({},n):n);return
        e},t.widget.bridge=function(e,i){var n=i.prototype.widgetFullName||e;t.fn[e]=function(o){var
        a="string"==typeof o,r=s.call(arguments,1),h=this;return
        a?this.length||"instance"!==o?this.each(function(){var
        i,s=t.data(this,n);return"instance"===o?(h=s,!1):s?t.isFunction(s[o])&&"_"!==o.charAt(0)?(i=s[o].apply(s,r),i!==s&&void
        0!==i?(h=i&&i.jquery?h.pushStack(i.get()):i,!1):void 0):t.error("no such method '"+o+"' for "+e+"
        widget instance"):t.error("cannot call methods on "+e+" prior to initialization; "+"attempted to call
        method '"+o+"')}):h=void
        0:(r.length&&(o=t.widget.extend.apply(null,[o].concat(r))),this.each(function(){var
        e=t.data(this,n);e?(e.option(o||{}),e._init&&e._init()):t.data(this,n,new
        i(o,this))})),h}},t.Widget=function(){},t.Widget._childConstructors=[],t.Widget.prototype={widgetName:"widget",widgetEv
        e=this;this._destroy(),t.each(this.classesElementLookup,function(t,i){e._removeClass(i,t)}),this.element.off(this.event
        this.element},option:function(e,i){var s,n,o,a=e;if(0===arguments.length)return
        t.widget.extend({},this.options);if("string"==typeof
        e)if(a={},s=e.split("."),e=s.shift(),s.length){for(n=a[e]=t.widget.extend({},this.options[e]),o=0;s.length-1>o;o++)n[s
        void 0===n[e]?null:n[e];n[e]=i}else{if(1===arguments.length)return void
        0===this.options[e]?null:this.options[e];a[e]=i}return
        this._setOptions(a),this},_setOptions:function(t){var e;for(e in t)this._setOption(e,t[e]);return
```

```
this},_setOption:function(t,e){return"classes"===t&&this._setOptionClasses(e),this.options[t]=e,"disabled"===t&&this._s
i,s,n;for(i in
e)n=this.classesElementLookup[i],e[i]!==this.options.classes[i]&&n&&n.length&&(s=t(n.get()),this._removeClass(n,i),s.ad
this._setOptions({disabled:!1})},disable:function(){return
this._setOptions({disabled:!0})},_classes:function(e){function i(i,o){var
a,r;for(r=0;i.length>r;r++)a=n.classesElementLookup[i[r]]||t(),a=e.add?t(t.unique(a.get().concat(e.element.get()))):t(a
s=[],n=this;return
e=t.extend({element:this.element,classes:this.options.classes||{}},e),this._on(e.element,{remove:"_untrackClassesElemen
")},_untrackClassesElement:function(e){var
i=this;t.each(i.classesElementLookup,function(s,n){-1!==t.inArray(e.target,n)&&(i.classesElementLookup[s]=t(n.not(e.tar
this._toggleClass(t,e,i,!1)},_addClass:function(t,e,i){return
this._toggleClass(t,e,i,!0)},_toggleClass:function(t,e,i,s){s="boolean"==typeof s?s:i;var
n="string"==typeof t||null===t,o={extra:n?e:i,keys:n?t:e,element:n?this.element:t,add:s};return
o.element.toggleClass(this._classes(o),s),this},_on:function(e,i,s){var n,o=this;"boolean"!=typeof
e&&(s=i,i=e,e=!1),s?(i=n=t(i),this.bindings=this.bindings.add(i)):(s=i,i=this.element,n=this.widget()),t.each(s,functio
r(){return e||o.options.disabled!==!0&&!t(this).hasClass("ui-state-disabled")?("string"==typeof
a?o[a]:a).apply(o,arguments):void 0}"string"!=typeof a&&(r.guid=a.guid=a.guid||r.guid||t.guid++);var
h=s.match(/^([\w:-]*)\s*(.*)$/),l=h[1]+o.eventNamespace,c=h[2];c?n.on(l,c,r):i.on(l,r)})},_off:function(e,i){i=(i||"")
").join(this.eventNamespace+"
")+this.eventNamespace,e.off(i).off(i),this.bindings=t(this.bindings.not(e).get()),this.focusable=t(this.focusable.not
i(){return("string"==typeof t?s[t]:t).apply(s,arguments)}var s=this;return
setTimeout(i,e||0)},_hoverable:function(e){this.hoverable=this.hoverable.add(e),this._on(e,{mouseenter:function(e){this
n,o,a=this.options[e];if(s=s||{},i=t.Event(i),i.type=(e===this.widgetEventPrefix?e:this.widgetEventPrefix+e).toLowerCas
in o)n in i||(i[n]=o[n]);return
this.element.trigger(i,s),!(t.isFunction(a)&&a.apply(this.element[0],[i].concat(s))===!1||i.isDefaultPrevented())}},t.e
n&&(n={effect:n});var a,r=n?n===!0||"number"==typeof n?i:n.effect||i:e;n=n||{},"number"==typeof
n&&(n={duration:n}),a=!t.isEmptyObject(n),n.complete=o,n.delay&&s.delay(n.delay),a&&t.effects&&t.effects.effect[r]?s[e]
e(t,e,i){return[parseFloat(t[0])*(u.test(t[0])?e/100:1),parseFloat(t[1])*(u.test(t[1])?i/100:1)]}function
i(e,i){return parseInt(t.css(e,i),10)||0}function s(e){var i=e[0];return
9===i.nodeType?{width:e.width(),height:e.height(),offset:{top:0,left:0}}:t.isWindow(i)?{width:e.width(),height:e.height
n,o=Math.max,a=Math.abs,r=/left|center|right/,h=/top|center|bottom/,l=/[\+\-]\d+(\.[\d]+)?%?/,c=/^\w+/,u=/%$/,d=t.fn.po
0!==n)return n;var e,i,s=t("<div
style='display:block;position:absolute;width:50px;height:50px;overflow:hidden;'><div
style='height:100px;width:auto;'></div></div>"),o=s.children()[0];return
t("body").append(s),e=o.offsetWidth,s.css("overflow","scroll"),i=o.offsetWidth,e===i&&(i=s[0].clientWidth),s.remove(),r
i=e.isWindow||e.isDocument?"":e.element.css("overflow-x"),s=e.isWindow||e.isDocument?"":e.element.css("overflow-y"),n="
i=t(e||window),s=t.isWindow(i[0]),n=!!i[0]&&9===i[0].nodeType,o=!s&&!n;return{element:i,isWindow:s,isDocument:n,offset
d.apply(this,arguments);n=t.extend({},n);var
u,p,f,m,g,_,v=t(n.of),b=t.position.getWithinInfo(n.within),y=t.position.getScrollInfo(b),w=(n.collision||"flip").split
"),x={};return _=s(v),v[0].preventDefault&&(n.at="left
top"),p=_.width,f=_.height,m=_.offset,g=t.extend({},m),t.each(["my","at"],function(){var
t,e,i=(n[this]||"").split("
");1===i.length&&(i=r.test(i[0])?i.concat(["center"]):h.test(i[0])?["center"].concat(i):["center","center"]),i[0]=r.tes
s,r,h=t(this),l=h.outerWidth(),c=h.outerHeight(),d=i(this,"marginLeft"),_=i(this,"marginTop"),k=l+d+i(this,"marginRight
e=m.left-D.left,i=e+p-l,s=m.top-D.top,r=s+f-c,u={target:{element:v,left:m.left,top:m.top,width:p,height:f},element:{ele
i,s=e.within,n=s.isWindow?s.scrollLeft:s.offset.left,a=s.width,r=t.left-e.collisionPosition.marginLeft,h=n-r,l=r+e.coll
i,s=e.within,n=s.isWindow?s.scrollTop:s.offset.top,a=e.within.height,r=t.top-e.collisionPosition.marginTop,h=n-r,l=r+e
i,s,n=e.within,o=n.offset.left+n.scrollLeft,r=n.width,h=n.isWindow?n.scrollLeft:n.offset.left,l=t.left-e.collisionPosit
i,s,n=e.within,o=n.offset.top+n.scrollTop,r=n.height,h=n.isWindow?n.scrollTop:n.offset.top,l=t.top-e.collisionPosition
function(i){return!!t.data(i,e)})):function(e,i,s){return!!t.data(e,s[3])})}),t.fn.extend({disableSelection:function(){v
t="onselectstart"in document.createElement("div")?"selectstart":"mousedown";return function(){return
this.on(t+".ui-disableSelection",function(t){t.preventDefault()})}}(),enableSelection:function(){return
this.off(".ui-disableSelection")}}),t.ui.focusable=function(i,s){var
n,o,a,r,h,l=i.nodeName.toLowerCase();return"area"===l?(n=i.parentNode,o=n.name,i.href&&o&&"map"===n.nodeName.toLowerCas
t.ui.focusable(e,null!=t.attr(e,"tabindex"))}}),t.ui.focusable,t.fn.form=function(){return"string"==typeof
this[0].form?this.closest("form"):t(this[0].form)},t.ui.formResetMixin={_formResetHandler:function(){var
e=t(this);setTimeout(function(){var
i=e.data("ui-form-reset-instances");t.each(i,function(){this.refresh()})})},_bindFormResetHandler:function(){if(this.fo
t=this.form.data("ui-form-reset-instances")||[];t.length||this.form.on("reset.ui-form-reset",this._formResetHandler),t
e=this.form.data("ui-form-reset-instances");e.splice(t.inArray(this,e),1),e.length?this.form.data("ui-form-reset-instan
s(e,i,s,o){return
t.each(n,function(){i-=parseFloat(t.css(e,"padding"+this))||0,s&&(i-=parseFloat(t.css(e,"border"+this+"Width"))||0),o&&
n="Width"===i?["Left","Right"]:["Top","Bottom"],o=i.toLowerCase(),a={innerWidth:t.fn.innerWidth,innerHeight:t.fn.innerH
void
0===e?a["inner"+i].call(this):this.each(function(){t(this).css(o,s(this,e)+"px")})},t.fn["outer"+i]=function(e,n){retur
e?a["outer"+i].call(this,e):this.each(function(){t(this).css(o,s(this,e,!0,n)+"px")})}}),t.fn.addBack=function(t){retur
this.add(null==t?this.prevObject:this.prevObject.filter(t))},t.ui.keyCode={BACKSPACE:8,COMMA:188,DELETE:46,DOWN:40,END
t=/([!"#$%&'()*+,./:;<=>?@[\]^`{|}~])/g;return function(e){return
e.replace(t,"\\$1")}}(),t.fn.labels=function(){var e,i,s,n,o;return
this[0].labels&&this[0].labels.length?this.pushStack(this[0].labels):(n=this.eq(0).parents("label"),s=this.attr("id"),s
i=this.css("position"),s="absolute"===i,n=e?/(auto|scroll|hidden)/:/(auto|scroll)/,o=this.parents().filter(function(){v
e=t(this);return
s&&"static"===e.css("position")?!1:n.test(e.css("overflow")+e.css("overflow-y")+e.css("overflow-x"))}).eq(0);return"fix
i=t.attr(e,"tabindex"),s=null!=i;return(!s||i>=0)&&t.ui.focusable(e,s)}}),t.fn.extend({uniqueId:function(){var
t=0;return function(){return this.each(function(){this.id||(this.id="ui-id-"+
++t)})}}(),removeUniqueId:function(){return
this.each(function(){/^ui-id-\d+$/.test(this.id)&&t(this).removeAttr("id")})}}),t.ui.ie=!!/msie
[\w.]+/.exec(navigator.userAgent.toLowerCase());var
n=!1;t(document).on("mouseup",function(){n=!1}),t.widget("ui.mouse",{version:"1.12.1",options:{cancel:"input,
textarea, button, select, option",distance:1,delay:0},_mouseInit:function(){var
e=this;this.element.on("mousedown."+this.widgetName,function(t){return
e._mouseDown(t)}).on("click."+this.widgetName,function(i){return!0===t.data(i.target,e.widgetName+".preventClickEvent")
0}),this.started=!1},_mouseDestroy:function(){this.element.off("."+this.widgetName),this._mouseMoveDelegate&&this.docum
i=this,s=1===e.which,o="string"==typeof
this.options.cancel&&e.target.nodeName?t(e.target).closest(this.options.cancel).length:!1;return
s&&!o&&this._mouseCapture(e)?(this.mouseDelayMet=!this.options.delay,this.mouseDelayMet||(this._mouseDelayTimer=setTime
i._mouseMove(t)},this._mouseUpDelegate=function(t){return
```

```
i._mouseUp(t)},this.document.on("mousemove."+this.widgetName,this._mouseMoveDelegate).on("mouseup."+this.widgetName,thi
this._mouseUp(e);if(!e.which)if(e.originalEvent.altKey||e.originalEvent.ctrlKey||e.originalEvent.metaKey||e.originalEve
if(!this.ignoreMissingWhich)return
this._mouseUp(e)}return(e.which||e.button)&&(this._mouseMoved=!0),this._mouseStarted?(this._mouseDrag(e),e.preventDefau
this._mouseDelayTimer),this.ignoreMissingWhich=!1,n=!1,e.preventDefault()},_mouseDistanceMet:function(t){return
Math.max(Math.abs(this._mouseDownEvent.pageX-t.pageX),Math.abs(this._mouseDownEvent.pageY-t.pageY))>=this.options.dista
this.mouseDelayMet},_mouseStart:function(){},_mouseDrag:function(){},_mouseStop:function(){},_mouseCapture:function(){r
n,o=t.ui[e].prototype;for(n in
s)o.plugins[n]=o.plugins[n]||[],o.plugins[n].push([i,s[n]])},call:function(t,e,i,s){var
n,o=t.plugins[e];if(o&&(s||t.element[0].parentNode&&11!==t.element[0].parentNode.nodeType))for(n=0;o.length>n;n++)t.opt
ui-icon-gripsmall-diagonal-se"},containment:!1,ghost:!1,grid:!1,handles:"e,s,se",helper:!1,maxHeight:null,maxWidth:null
parseFloat(t)||0},_isNumber:function(t){return!isNaN(parseFloat(t))},_hasScroll:function(e,i){if("hidden"===t(e).css("o
s=i&&"left"===i?"scrollLeft":"scrollTop",n=!1;return
e[s]>0?!0:(e[s]=1,n=e[s]>0,e[s]=0,n)},_create:function(){var
e,i=this.options,s=this;this._addClass("ui-resizable"),t.extend(this,{_aspectRatio:!!i.aspectRatio,aspectRatio:i.aspect
class='ui-wrapper' style='overflow:
hidden;'></div>").css({position:this.element.css("position"),width:this.element.outerWidth(),height:this.element.outerH
e,i=function(e){t(e).removeData("resizable").removeData("ui-resizable").off(".resizable").find(".ui-resizable-handle")
this.elementIsWrapper&&(i(this.element),e=this.element,this.originalElement.css({position:e.css("position"),width:e.out
e,i,s,n,o,a=this.options,r=this;if(this.handles=a.handles||(t(".ui-resizable-handle",this.element).length?{n:".ui-resiz
"+n),o.css({zIndex:a.zIndex}),this.handles[e]=".ui-resizable-"+e,this.element.append(o);this._renderAxis=function(e){va
i,s,n,o;e=e||this.element;for(i in
this.handles)this.handles[i].constructor===String?this.handles[i]=this.element.children(this.handles[i]).first().show()
i,s,n=!1;for(i in
this.handles)s=t(this.handles[i])[0],(s===e.target||t.contains(s,e.target))&&(n=!0);return!this.options.disabled&&n},_r
i,s,n,o=this.options,a=this.element;return
this.resizing=!0,this._renderProxy(),i=this._num(this.helper.css("left")),s=this._num(this.helper.css("top")),o.contain
o.aspectRatio?o.aspectRatio:this.originalSize.width/this.originalSize.height||1,n=t(".ui-resizable-"+this.axis).css("cu
i,s,n=this.originalMousePosition,o=this.axis,a=e.pageX-n.left||0,r=e.pageY-n.top||0,h=this._change[o];return
this._updatePrevProperties(),h?(i=h.apply(this,[e,a,r]),this._updateVirtualBoundaries(e.shiftKey),(this._aspectRatio||e
i,s,n,o,a,r,h,l=this.options,c=this;return
this._helper&&(i=this._proportionallyResizeElements,s=i.length&&/textarea/i.test(i[0].nodeName),n=s&&this._hasScroll(i
t={};return
this.position.top!==this.prevPosition.top&&(t.top=this.position.top+"px"),this.position.left!==this.prevPosition.left&&
e,i,s,n,o,a=this.options;o={minWidth:this._isNumber(a.minWidth)?a.minWidth:0,maxWidth:this._isNumber(a.maxWidth)?a.maxW
e=this.position,i=this.size,s=this.axis;return
this._isNumber(t.height)?t.width=t.height*this.aspectRatio:this._isNumber(t.width)&&(t.height=t.width/this.aspectRatio)
e=this._vBoundaries,i=this.axis,s=this._isNumber(t.width)&&e.maxWidth&&e.maxWidth<t.width,n=this._isNumber(t.height)&&e
o&&(t.width=e.minWidth),a&&(t.height=e.minHeight),s&&(t.width=e.maxWidth),n&&(t.height=e.maxHeight),o&&l&&(t.left=r-e.m
e=0,i=[],s=[t.css("borderTopWidth"),t.css("borderRightWidth"),t.css("borderBottomWidth"),t.css("borderLeftWidth")],n=[t
t,e=0,i=this.helper||this.element;this._proportionallyResizeElements.length>e;e++)t=this._proportionallyResizeElements
e=this.element,i=this.options;this.elementOffset=e.offset(),this._helper?(this.helper=this.helper||t("<div
style='overflow:hidden;'></div>"),this._addClass(this.helper,this._helper),this.helper.css({width:this.element.outerWid
},_change:{e:function(t,e){return{width:this.originalSize.width+e}},w:function(t,e){var
i=this.originalSize,s=this.originalPosition;return{left:s.left+e,width:i.width-e}},n:function(t,e,i){var
s=this.originalSize,n=this.originalPosition;return{top:n.top+i,height:s.height-i}},s:function(t,e,i){return{height:this
t.extend(this._change.s.apply(this,arguments),this._change.e.apply(this,[e,i,s]))},sw:function(e,i,s){return
t.extend(this._change.s.apply(this,arguments),this._change.w.apply(this,[e,i,s]))},ne:function(e,i,s){return
t.extend(this._change.n.apply(this,arguments),this._change.e.apply(this,[e,i,s]))},nw:function(e,i,s){return
t.extend(this._change.n.apply(this,arguments),this._change.w.apply(this,[e,i,s]))}},_propagate:function(e,i){t.ui.plugi
i=t(this).resizable("instance"),s=i.options,n=i._proportionallyResizeElements,o=n.length&&/textarea/i.test(n[0].nodeNam
s={width:parseFloat(i.element.css("width")),height:parseFloat(i.element.css("height")),top:parseFloat(i.element.css("to
e,i,s,n,o,a,r,h=t(this).resizable("instance"),l=h.options,c=h.element,u=l.containment,d=u instanceof
t?u.get(0):/parent/.test(u)?c.parent().get(0):u;d&&(h.containerElement=t(d),/document/.test(u)||u===document?(h.contain
i,s,n,o,a=t(this).resizable("instance"),r=a.options,h=a.containerOffset,l=a.position,c=a._aspectRatio||e.shiftKey,u={to
e=t(this).resizable("instance"),i=e.options,s=e.containerOffset,n=e.containerPosition,o=e.containerElement,a=t(e.helper
e=t(this).resizable("instance"),i=e.options;t(i.alsoResize).each(function(){var
e=t(this);e.data("ui-resizable-alsoresize",{width:parseFloat(e.width()),height:parseFloat(e.height()),left:parseFloat(e
s=t(this).resizable("instance"),n=s.options,o=s.originalSize,a=s.originalPosition,r={height:s.size.height-o.height||0,w
e=t(this),s=t(this).data("ui-resizable-alsoresize"),n={},o=e.parents(i.originalElement[0]).length?["width","height"]:["
i=(s[e]||0)+(r[e]||0);i&&i>=0&&(n[e]=i||null)},e.css(n)})},stop:function(){t(this).removeData("ui-resizable-alsoresize
e=t(this).resizable("instance"),i=e.size;e.ghost=e.originalElement.clone(),e.ghost.css({opacity:.25,display:"block",pos
e.options.ghost&&e.ghost.addClass(this.options.ghost),e.ghost.appendTo(e.helper)},resize:function(){var
e=t(this).resizable("instance");e.ghost&&e.ghost.css({position:"relative",height:e.size.height,width:e.size.width})},st
e=t(this).resizable("instance");e.ghost&&e.helper&&e.helper.get(0).removeChild(e.ghost.get(0))}})},t.ui.plugin.add("resi
e,i=t(this).resizable("instance"),s=i.options,n=i.size,o=i.originalSize,a=i.originalPosition,r=i.axis,h="number"==typeo
s.grid?[s.grid,s.grid]:s.grid,l=h[0]||1,c=h[1]||1,u=Math.round((n.width-o.width)/l)*l,d=Math.round((n.height-o.height)/
 * Copyright (c) 2007 Ariel Flesler - aflesler  gmail • com | https://github.com/flesler
 * Licensed under MIT
 * @author Ariel Flesler
 * @version 2.1.2
 */
;(function(f){"use strict";"function"===typeof
define&&define.amd?define(["jquery"],f):"undefined"!==typeof
module&&module.exports?module.exports=f(require("jquery")):f(jQuery)})(function($){"use
strict";function
n(a){return!a.nodeName||-1!==$.inArray(a.nodeName.toLowerCase(),["iframe","#document","html","body"])}function
h(a){return $.isFunction(a)||$.isPlainObject(a)?a:{top:a,left:a}}var
p=$.scrollTo=function(a,d,b){return
$(window).scrollTo(a,d,b)};p.defaults={axis:"xy",duration:0,limit:!0};$.fn.scrollTo=function(a,d,b){"object"===
typeof d&&(b=d,d=0);"function"===typeof
b&&(b={onAfter:b});"max"===a&&(a=9E9);b=$.extend({},p.defaults,b);d=d||b.duration;var
u=b.queue&&1<b.axis.length;u&&(d/=2);b.offset=h(b.offset);b.over=h(b.over);return
this.each(function(){function k(a){var
k=$.extend({},b,{queue:!0,duration:d,complete:a&&function(){a.call(q,e,b)}});r.animate(f,k)}if(null!==a){var
l=n(this),q=l?this.contentWindow||window:this,r=$(q),e=a,f={},t;switch(typeof e){case "number":case
"string":if(/^([+-]=?)?\d+(\.\d+)?(px|%)?$/.test(e)){e= h(e);break}e=l?$(e):$(e,q);case
```

```
"object":if(e.length===0)return;if(e.is||e.style)t=(e=$(e)).offset()}var
v=$.isFunction(b.offset)&&b.offset(q,e)||b.offset;$.each(b.axis.split(""),function(a,c){var
d="x"===c?"Left":"Top",m=d.toLowerCase(),g="scroll"+d,h=r[g](),n=p.max(q,c);t?(f[g]=t[m]+(l?0:h-r.offset()[m]),b.margin
"%"===d.slice(-1)?parseFloat(d)/100*n:d);b.limit&&/^\d+$/.test(f[g])&&(f[g]=0>=f[g]?0:Math.min(f[g],n));!a&&1<b.axis.le
b="x"===d?"Width":"Height",h="scroll"+b;if(!n(a))return a[h]-$(a)[b.toLowerCase()]();var
b="client"+b,k=a.ownerDocument||a.document,l=k.documentElement,k=k.body;return
Math.max(l[h],k[h])-Math.min(l[b],k[b])};$.Tween.propHooks.scrollLeft=$.Tween.propHooks.scrollTop={get:function(a){retu
$(a.elem)[a.prop]()}, set:function(a){var
d=this.get(a);if(a.options.interrupt&&a._last&&a._last!==d)return $(a.elem).stop();var
b=Math.round(a.now);d!==b&&($(a.elem)[a.prop](b),a._last=this.get(a))}};return p});
/*!
 PowerTip v1.3.1 (2018-04-15)
 https://stevenbenner.github.io/jquery-powertip/
 Copyright (c) 2018 Steven Benner (http://stevenbenner.com/).
 Released under MIT license.
 https://raw.github.com/stevenbenner/jquery-powertip/master/LICENSE.txt
*/
(function(root,factory){if(typeof define===="function"&&define.amd){define(["jquery"],factory)}else
if(typeof
module===="object"&&module.exports){module.exports=factory(require("jquery"))}else{factory(root.jQuery)}})(this,function
$document=$(document),$window=$(window),$body=$("body");var
DATA_DISPLAYCONTROLLER="displayController",DATA_HASACTIVEHOVER="hasActiveHover",DATA_FORCEDOPEN="forcedOpen",DATA_HASMO
session={tooltips:null,isTipOpen:false,isFixedTipOpen:false,isClosing:false,tipOpenImminent:false,activeHover:null,curr
Collision={none:0,top:1,bottom:2,left:4,right:8};$.fn.powerTip=function(opts,arg){var
targetElements=this,options,tipController;if(!targetElements.length){return
targetElements}if($.type(opts)===="string"&&$.powerTip[opts]){return
$.powerTip[opts].call(targetElements,targetElements,arg)}options=$.extend({},$.fn.powerTip.defaults,opts);tipController
TooltipController(options);initTracking();targetElements.each(function elementSetup(){var
$this=$(this),dataPowertip=$this.data(DATA_POWERTIP),dataElem=$this.data(DATA_POWERTIPJQ),dataTarget=$this.data(DATA_PO
DisplayController($this,options,tipController)});if(!options.manual){$.each(options.openEvents,function(idx,evt){if($
elementToggle(event){$.powerTip.toggle(this,event)})}else{targetElements.on(evt+EVENT_NAMESPACE,function
elementOpen(event){$.powerTip.show(this,event)})}});$.each(options.closeEvents,function(idx,evt){if($.inArray(evt,optio
elementClose(event){$.powerTip.hide(this,!isMouseEvent(event))})}});targetElements.on("keydown"+EVENT_NAMESPACE,functio
elementKeyDown(event){if(event.keyCode===27){$.powerTip.hide(this,true)}})}return
targetElements};$.fn.powerTip.defaults={fadeInTime:200,fadeOutTime:100,followMouse:false,popupId:"powerTip",popupClass
apiShowTip(element,event){if(isMouseEvent(event)){trackMouse(event);session.previousX=event.pageX;session.previousY=eve
element},reposition:function
apiResetPosition(element){$(element).first().data(DATA_DISPLAYCONTROLLER).resetPosition();return
element},hide:function apiCloseTip(element,immediate){var
displayController;immediate=element?immediate:true;if(element){displayController=$(element).first().data(DATA_DISPLAYCO
if(session.activeHover){displayController=session.activeHover.data(DATA_DISPLAYCONTROLLER)}if(displayController){displa
element},toggle:function
apiToggle(element,event){if(session.activeHover&&session.activeHover.is(element)){$.powerTip.hide(element,!isMouseEvent
element}};$.powerTip.showTip=$.powerTip.show;$.powerTip.closeTip=$.powerTip.hide;function
CSSCoordinates(){var
me=this;me.top="auto";me.left="auto";me.right="auto";me.bottom="auto";me.set=function(property,value){if($.isNumeric(va
DisplayController(element,options,tipController){var hoverTimer=null,myCloseDelay=null;function
openTooltip(immediate,forceOpen){cancelTimer();if(!element.data(DATA_HASACTIVEHOVER)){if(!immediate){session.tipOpenImm
intentDelay(){hoverTimer=null;checkForIntent()},options.intentPollInterval)}else{if(forceOpen){element.data(DATA_FORCED
closeTooltip(disableDelay){if(myCloseDelay){myCloseDelay=session.closeDelayTimeout=clearTimeout(myCloseDelay);session.c
closeDelay(){session.closeDelayTimeout=null;tipController.hideTip(element);session.delayInProgress=false;myCloseDelay=n
checkForIntent(){var
xDifference=Math.abs(session.previousX-session.currentX),yDifference=Math.abs(session.previousY-session.currentY),total
cancelTimer(stopClose){hoverTimer=clearTimeout(hoverTimer);if(session.closeDelayTimeout&&myCloseDelay===session.closeDe
cancelClose(){session.closeDelayTimeout=clearTimeout(session.closeDelayTimeout);session.delayInProgress=false}function
closeAnyDelayed(){if(session.delayInProgress&&session.activeHover&&!session.activeHover.is(element)){session.activeHove
repositionTooltip(){tipController.resetPosition(element)}this.show=openTooltip;this.hide=closeTooltip;this.cancel=cance
PlacementCalculator(){function
computePlacementCoords(element,placement,tipWidth,tipHeight,offset){var
placementBase=placement.split("-")[0],coords=new
CSSCoordinates,position;if(isSvgElement(element)){position=getSvgPlacement(element,placementBase)}else{position=getHtml
coords}function getHtmlPlacement(element,placement){var
objectOffset=element.offset(),objectWidth=element.outerWidth(),objectHeight=element.outerHeight(),left,top;switch(place
getSvgPlacement(element,placement){var
svgElement=element.closest("svg")[0],domElement=element[0],point=svgElement.createSVGPoint(),boundingBox=domElement.get
pushPlacement(){placements.push(point.matrixTransform(matrix))}point.x=boundingBox.x;point.y=boundingBox.y;pushPlacemen
TooltipController(options){var placementCalculator=new
PlacementCalculator,tipElement=$("#"+options.popupId);if(tipElement.length===0){tipElement=$("<div/>",{id:options.popup
beginShowTip(element){element.data(DATA_HASACTIVEHOVER,true);tipElement.queue(function
queueTipInit(next){showTip(element);next()})}function showTip(element){var
tipContent;if(!element.data(DATA_HASACTIVEHOVER)){return}if(session.isTipOpen){if(!session.isClosing){hideTip(session.a
queueTipAgain(next){showTip(element);next()});return}element.trigger("powerTipPreRender");tipContent=getTooltipContent
documentClick(event){var
target=event.target;if(target!==element[0]){if(options.mouseOnToPopup){if(target!==tipElement[0]&&!$.contains(tipElemen
tipMouseEnter(){if(session.activeHover){session.activeHover.data(DATA_DISPLAYCONTROLLER).cancel()}});tipElement.on("mou
tipMouseLeave(){if(session.activeHover){session.activeHover.data(DATA_DISPLAYCONTROLLER).hide()}})}tipElement.fadeIn(op
fadeInCallback(){if(!session.desyncTimeout){session.desyncTimeout=setInterval(closeDesyncedTip,500)}element.trigger("po
hideTip(element){session.isClosing=true;session.isTipOpen=false;session.desyncTimeout=clearInterval(session.desyncTime
fadeOutCallback(){var coords=new
CSSCoordinates;session.activeHover=null;session.isClosing=false;session.isFixedTipOpen=false;tipElement.removeClass();c
positionTipOnCursor(){var
tipWidth,tipHeight,coords,collisions,collisionCount;if(!session.isFixedTipOpen&&(session.isTipOpen||session.tipOpenImm
CSSCoordinates;coords.set("top",session.currentY+options.offset);coords.set("left",session.currentX+options.offset);col
if(collisions===Collision.bottom){coords.set("top",session.scrollTop+session.windowHeight-tipHeight)}}else{coords.set("
positionTipOnElement(element){var
priorityList,finalPlacement;if(options.smartPlacement||options.followMouse&&element.data(DATA_FORCEDOPEN)){priorityList
collisions=getViewportCollisions(placeTooltip(element,pos),tipElement.outerWidth(),tipElement.outerHeight());finalPlace
```

```
collisions!==Collision.none})}else{placeTooltip(element,options.placement);finalPlacement=options.placement}tipElement
nw sw e ne se n s w se-alt sw-alt ne-alt nw-alt");tipElement.addClass(finalPlacement)}function
placeTooltip(element,placement){var iterationCount=0,tipWidth,tipHeight,coords=new
CSSCoordinates;coords.set("top",0);coords.set("left",0);tipElement.css(coords);do{tipWidth=tipElement.outerWidth();tipH
coords}function closeDesyncedTip(){var
isDesynced=false,hasDesyncableCloseEvent=$.grep(["mouseleave","mouseout","blur","focusout"],function(eventType){return
$.inArray(eventType,options.closeEvents)!==-1}).length>0;if(session.isTipOpen&&!session.isClosing&&!session.delayInProc
if(!isMouseOver(session.activeHover)&&!session.activeHover.is(":focus")&&!session.activeHover.data(DATA_FORCEDOPEN)){i
isSvgElement(element){return Boolean(window.SVGElement&&element[0]instanceof SVGElement)}function
isMouseEvent(event){return Boolean(event&&$.inArray(event.type,MOUSE_EVENTS)>-1&&typeof
event.pageX=="number")}function
initTracking(){if(!session.mouseTrackingActive){session.mouseTrackingActive=true;getViewportDimensions();$(getViewportD
getViewportDimensions(){session.scrollLeft=$window.scrollLeft();session.scrollTop=$window.scrollTop();session.windowWid
trackResize(){session.windowWidth=$window.width();session.windowHeight=$window.height()}function
trackScroll(){var
x=$window.scrollLeft(),y=$window.scrollTop();if(x!==session.scrollLeft){session.currentX+=x-session.scrollLeft;session
trackMouse(event){session.currentX=event.pageX;session.currentY=event.pageY}function
isMouseOver(element){var
elementPosition=element.offset(),elementBox=element[0].getBoundingClientRect(),elementWidth=elementBox.right-elementBox
session.currentX>=elementPosition.left&&session.currentX<=elementPosition.left+elementWidth&&session.currentY>=elementP
getTooltipContent(element){var
tipText=element.data(DATA_POWERTIP),tipObject=element.data(DATA_POWERTIPJQ),tipTarget=element.data(DATA_POWERTIPTARGET)
if(tipObject){if($.isFunction(tipObject)){tipObject=tipObject.call(element[0])}if(tipObject.length>0){content=tipObject
if(tipTarget){targetElement=$("#"+tipTarget);if(targetElement.length>0){content=targetElement.html()}}return
content}function getViewportCollisions(coords,elementWidth,elementHeight){var
viewportTop=session.scrollTop,viewportLeft=session.scrollLeft,viewportBottom=viewportTop+session.windowHeight,viewportR
collisions}function countFlags(value){var count=0;while(value){value&=value-1;count++}return
count}return $.powerTip});/*!
 * jQuery UI Touch Punch 0.2.3
 *
 * Copyright 2011-2014, Dave Furfero
 * Dual licensed under the MIT or GPL Version 2 licenses.
 *
 * Depends:
 *  jquery.ui.widget.js
 *  jquery.ui.mouse.js
 */
!function(a){function f(a,b){if(!(a.originalEvent.touches.length>1)){a.preventDefault();var
c=a.originalEvent.changedTouches[0],d=document.createEvent("MouseEvents");d.initMouseEvent(b,!0,!0,window,1,c.screenX,c
document,a.support.touch){var
e,b=a.ui.mouse.prototype,c=b._mouseInit,d=b._mouseDestroy;b._touchStart=function(a){var
b=this;!e&&b._mouseCapture(a.originalEvent.changedTouches[0])&&(e=!0,b._touchMoved=!1,f(a,"mouseover"),f(a,"mousemove")
b=this;b.element.bind({touchstart:a.proxy(b,"_touchStart"),touchmove:a.proxy(b,"_touchMove"),touchend:a.proxy(b,"_touch
b=this;b.element.unbind({touchstart:a.proxy(b,"_touchStart"),touchmove:a.proxy(b,"_touchMove"),touchend:a.proxy(b,"_tou
SmartMenus jQuery Plugin - v1.1.0 - September 17, 2017
 * http://www.smartmenus.org/
 * Copyright Vasil Dinkov, Vadikom Web Ltd. http://vadikom.com; Licensed MIT
*/(function(t){"function"==typeof define&&define.amd?define(["jquery"],t):"object"==typeof
module&&"object"==typeof
module.exports?module.exports=t(require("jquery")):t(jQuery)})(function($){function
initMouseDetection(t){var
e=".smartmenus_mouse";if(mouseDetectionEnabled||t)mouseDetectionEnabled&&t&&($(document).off(e),mouseDetectionEnabled=
i=!0,s=null,o={mousemove:function(t){var e={x:t.pageX,y:t.pageY,timeStamp:(new
Date).getTime()};if(s){var
o=Math.abs(s.x-e.x),a=Math.abs(s.y-e.y);if((o>0||a>0)&&2>=o&&2>=a&&300>=e.timeStamp-s.timeStamp&&(mouse=!0,i)){var
n=$(t.target).closest("a");n.is("a")&&$.each(menuTrees,function(){return
$.contains(this.$root[0],n[0])?(this.itemEnter({currentTarget:n[0]}),!1):void
0}),i=!1}}s=e}};o[touchEvents?"touchstart":"pointerover pointermove pointerout MSPointerOver
MSPointerMove
MSPointerOut"]=function(t){isTouchEvent(t.originalEvent)&&(mouse=!1)},$(document).on(getEventsNS(o,e)),mouseDetectionEn
isTouchEvent(t){return!/^(4|mouse)$/.test(t.pointerType)}function getEventsNS(t,e){e||(e="");var
i={};for(var s in t)i[s.split(" ").join(e+" ")+e]=t[s];return i}var
menuTrees=[],mouse=!1,touchEvents="ontouchstart"in
window,mouseDetectionEnabled=!1,requestAnimationFrame=window.requestAnimationFrame||function(t){return
setTimeout(t,1e3/60)},cancelAnimationFrame=window.cancelAnimationFrame||function(t){clearTimeout(t)},canAnimate=!!$.fn
$.SmartMenus=function(t,e){this.$root=$(t),this.opts=e,this.rootId="",this.accessIdPrefix="",this.$subArrow=null,this.a
t.style||"webkitPerspective"in
t.style,this.wasCollapsible=!1,this.init()},$.extend($.SmartMenus,{hideAll:function(){$.each(menuTrees,function(){this
e=this;if(!t){menuTrees.push(this),this.rootId=((new
Date).getTime()+Math.random()+"").replace(/\D/g,""),this.accessIdPrefix="sm-"+this.rootId+"-",this.$root.hasClass("sm-
i=".smartmenus";this.$root.data("smartmenus",this).attr("data-smartmenus-id",this.rootId).dataSM("level",1).on(getEvent
focusin":$.proxy(this.rootOver,this),"mouseout
focusout":$.proxy(this.rootOut,this),keydown:$.proxy(this.rootKeyDown,this)},i)).on(getEventsNS({mouseenter:$.proxy(th
orientationchange":$.proxy(this.winResize,this)},i)),this.opts.subIndicators&&(this.$subArrow=$("<span/>").addClass("su
s=/(index|default)\.[^#\?\/]*/i,o=/#.*/,a=window.location.href.replace(s,""),n=a.replace(o,"");this.$root.find("a").eac
t=this.href.replace(s,""),i=$(this);(t==a||t==n)&&(i.addClass("current"),e.opts.markCurrentTree&&i.parentsUntil("[data-
e=".smartmenus";this.$root.removeData("smartmenus").removeAttr("data-smartmenus-id").removeDataSM("level").off(e),e+=th
i=this;this.$root.find("ul").each(function(){var
t=$(this);t.dataSM("scroll-arrows")&&t.dataSM("scroll-arrows").remove(),t.dataSM("shown-before")&&((i.opts.subMenusMinW
t=$(this);0==t.attr("id").indexOf(i.accessIdPrefix)&&t.removeAttr("id")}).removeClass("has-submenu").removeDataSM("sub
e=this.$root.offset();this.$disableOverlay=$('<div
class="sm-jquery-disable-overlay"/>').css({position:"absolute",top:e.top,left:e.left,width:this.$root.outerWidth(),heig
this.$touchScrollingSub?(this.$touchScrollingSub=null,void
0):((this.visibleSubMenus.length&&!$.contains(this.$root[0],t.target)||$(t.target).closest("a").length)&&this.menuHideA
0)},docTouchEnd:function(){if(this.lastTouch){if(!(!this.visibleSubMenus.length||void
0!==this.lastTouch.x2&&this.lastTouch.x1!=this.lastTouch.x2||void
0!==this.lastTouch.y2&&this.lastTouch.y1!=this.lastTouch.y2||this.lastTouch.target&&$.contains(this.$root[0],this.last
```

```
t=this;this.hideTimeout=setTimeout(function(){t.menuHideAll()},350)}this.lastTouch=null}},docTouchMove:function(t){if(t
e=t.originalEvent.touches[0];this.lastTouch.x2=e.pageX,this.lastTouch.y2=e.pageY}},docTouchStart:function(t){var
e=t.originalEvent.touches[0];this.lastTouch={x1:e.pageX,y1:e.pageY,target:e.target}},enable:function(){this.disabled&&
e=$(t).closest("ul");e.dataSM("in-mega");)e=e.parent().closest("ul");return
e[0]||null},getHeight:function(t){return this.getOffset(t,!0)},getOffset:function(t,e){var
i;"none"==t.css("display")&&(i={position:t[0].style.position,visibility:t[0].style.visibility},t.css({position:"absolut
s=t[0].getBoundingClientRect&&t[0].getBoundingClientRect(),o=s&&(e?s.height||s.bottom-s.top:s.width||s.right-s.left);re
o||0===o||(o=e?t[0].offsetHeight:t[0].offsetWidth),i&&t.hide().css(i),o},getStartZIndex:function(t){var
e=parseInt(this[t?"$root":"$firstSub"].css("z-index"));return!t&&isNaN(e)&&(e=parseInt(this.$root.css("z-index"))),isNa
t.touches&&t.touches[0]||t.changedTouches&&t.changedTouches[0]||t},getViewport:function(t){var
e=t?"Height":"Width",i=document.documentElement["client"+e],s=window["inner"+e];return
s&&(i=Math.min(i,s)),i},getViewportHeight:function(){return
this.getViewport(!0)},getViewportWidth:function(){return
this.getViewport()},getWidth:function(t){return
this.getOffset(t)},handleEvents:function(){return!this.disabled&&this.isCSSOn()},handleItemEvents:function(t){return
this.handleEvents()&&!this.isLinkInMegaMenu(t)},isCollapsible:function(){return"static"==this.$firstSub.css("position")
t="fixed"==this.$root.css("position");return
t||this.$root.parentsUntil("body").each(function(){return"fixed"==$(this).css("position")?(t=!0,!1):void
0}),t},isLinkInMegaMenu:function(t){return
$(this.getClosestMenu(t[0])).hasClass("mega-menu")},isTouchMode:function(){return!mouse||this.opts.noMouseOver||this.is
i=t.closest("ul"),s=i.dataSM("level");if(s>1&&(!this.activatedItems[s-2]||this.activatedItems[s-2][0]!=i.dataSM("parent
o=this;$(i.parentsUntil("[data-smartmenus-id]","ul").get().reverse()).add(i).each(function(){o.itemActivate($(this).dat
a=t.dataSM("sub");a&&(this.isTouchMode()||!this.opts.showOnClick||this.clickActivated)&&this.menuShow(a)}},itemBlur:fun
e=$(t.currentTarget);this.handleItemEvents(e)&&this.$root.triggerHandler("blur.smapi",e[0])},itemClick:function(t){var
e=$(t.currentTarget);if(this.handleItemEvents(e)){if(this.$touchScrollingSub&&this.$touchScrollingSub[0]==e.closest("ul
this.$touchScrollingSub=null,t.stopPropagation(),!1;if(this.$root.triggerHandler("click.smapi",e[0])===!1)return!1;var
i=$(t.target).is(".sub-arrow"),s=e.dataSM("sub"),o=s?2==s.dataSM("level"):!1,a=this.isCollapsible(),n=/toggle$/.test(th
this.focusActivated=!0,!1}else if(a&&(n||i))return
this.itemActivate(e,h),this.menuHide(s),n&&(this.focusActivated=!1),!1;return
this.opts.showOnClick&&o||e.hasClass("disabled")||this.$root.triggerHandler("select.smapi",e[0])===!1?!1:void
0}},itemDown:function(t){var
e=$(t.currentTarget);this.handleItemEvents(e)&&e.dataSM("mousedown",!0)},itemEnter:function(t){var
e=$(t.currentTarget);if(this.handleItemEvents(e)){if(!this.isTouchMode()){this.showTimeout&&(clearTimeout(this.showTime
i=this;this.showTimeout=setTimeout(function(){i.itemActivate(e)},this.opts.showOnClick&&1==e.closest("ul").dataSM("leve
e=$(t.currentTarget);this.handleItemEvents(e)&&(!this.focusActivated||this.isTouchMode()&&e.dataSM("mousedown")||this.a
e=$(t.currentTarget);this.handleItemEvents(e)&&(this.isTouchMode()||(e[0].blur(),this.showTimeout&&(clearTimeout(this.s
e=function(){t.css("z-index","")};this.isCollapsible()?canAnimate&&this.opts.collapsibleHideFunction?this.opts.collapsi
i=t.dataSM("level");this.activatedItems.splice(i-1,1),this.visibleSubMenus.splice($.inArray(t,this.visibleSubMenus),1),
t=this.opts.isPopup?1:0,e=this.visibleSubMenus.length-1;e>=t;e--)this.menuHide(this.visibleSubMenus[e]);this.opts.isPop
e=this.activatedItems.length-1;e>=t;e--){var
i=this.activatedItems[e].dataSM("sub");i&&this.menuHide(i)}},menuInit:function(t){if(!t.dataSM("in-mega")){t.hasClass("
e=2,i=t[0];(i=i.parentNode.parentNode)!=this.$root[0];)e++;var
s=t.prevAll("a").eq(-1);s.length||(s=t.prevAll().find("a").eq(-1)),s.addClass("has-submenu").dataSM("sub",t),t.dataSM("
o=s.attr("id")||this.accessIdPrefix+ ++this.idInc,a=t.attr("id")||this.accessIdPrefix+
++this.idInc;s.attr({id:o,"aria-haspopup":"true","aria-controls":a,"aria-expanded":"false"}),t.attr({id:a,role:"group",
e,i,s=t.dataSM("parent-a"),o=s.closest("li"),a=o.parent(),n=t.dataSM("level"),r=this.getWidth(t),h=this.getHeight(t),u=
y=l+e,I=c+i;if(M&&f>y?e=g?f-y+e:d-w:!M&&y+r>f+b&&(e=g?f+b-r-y+e:w-r),g||(S>h&&I+h>v+S?i+=v+S-h-I:(h>=S||v>I)&&(i+=v-I)
x=this;t.dataSM("scroll-arrows")||t.dataSM("scroll-arrows",$([$('<span class="scroll-up"><span
class="scroll-up-arrow"></span></span>')[0],$('<span class="scroll-down"><span
class="scroll-down-arrow"></span></span>')[0]]).on({mouseenter:function(){t.dataSM("scroll").up=$(this).hasClass("scrol
DOMMouseScroll":function(t){t.preventDefault()}}).insertAfter(t));var
A=".smartmenus_scroll";if(t.dataSM("scroll",{y:this.cssTransforms3d?0:i-m,step:1,itemH:m,subH:h,arrowDownH:this.getHei
DOMMouseScroll":function(e){x.menuScrollMousewheel(t,e)}},A)).dataSM("scroll-arrows").css({top:"auto",left:"0",marginLe
C={};C[touchEvents?"touchstart touchmove touchend":"pointerdown pointermove pointerup MSPointerDown
MSPointerMove
MSPointerUp"]=function(e){x.menuScrollTouch(t,e)},t.css({"touch-action":"none","-ms-touch-action":"none"}).on(getEvents
s,o=t.dataSM("scroll"),a=t.dataSM("scroll-arrows"),n=o.up?o.upEnd:o.downEnd;if(!e&&o.momentum){if(o.momentum*=.92,s=o.
this.menuScrollStop(t),void 0}else
s=i||(e||!this.opts.scrollAccelerate?this.opts.scrollStep:Math.floor(o.step));var
r=t.dataSM("level");if(this.activatedItems[r-1]&&this.activatedItems[r-1].dataSM("sub")&&this.activatedItems[r-1].dataS
"+o.y+"px, 0)",transform:"translate3d(0, "+o.y+"px,
0)"}:{marginTop:o.y}),mouse&&(o.up&&o.y>o.downEnd||!o.up&&o.y<o.upEnd)&&a.eq(o.up?1:0).show(),o.y==n)mouse&&a.eq(o.up?0
if(!e){this.opts.scrollAccelerate&&o.step<this.opts.scrollStep&&(o.step+=.2);var
h=this;this.scrollTimeout=requestAnimationFrame(function(){h.menuScroll(t)})}},menuScrollMousewheel:function(t,e){if(th
i=(e.wheelDelta||-e.detail)>0;t.dataSM("scroll-arrows").eq(i?0:1).is(":visible")&&(t.dataSM("scroll").up=i,this.menuScr
i=t.dataSM("scroll"),s=$(window).scrollTop()-t.dataSM("parent-a").offset().top-i.itemH;t.dataSM("scroll-arrows").eq(0)
e=t.dataSM("scroll"),i=$(window).scrollTop()-t.dataSM("parent-a").offset().top-e.itemH;this.cssTransforms3d&&(i=-(parse
this.scrollTimeout?(cancelAnimationFrame(this.scrollTimeout),this.scrollTimeout=0,t.dataSM("scroll").step=1,!0):void
0},menuScrollTouch:function(t,e){if(e=e.originalEvent,isTouchEvent(e)){var
i=this.getTouchPoint(e);if(this.getClosestMenu(i.target)==t[0]){var
s=t.dataSM("scroll");if(/(start|down)$/i.test(e.type))this.menuScrollStop(t)?(e.preventDefault(),this.$touchScrollingSu
if(/move$/i.test(e.type)){var o=void 0!==s.touchY?s.touchY:s.touchStartY;if(void
0!==o&&o!=i.pageY){this.$touchScrollingSub=t;var a=i.pageY>o;void
0!==s.up&&s.up!=a&&$.extend(s,{touchStartY:i.pageY,touchStartTime:e.timeStamp}),$.extend(s,{up:a,touchY:i.pageY}),this
void
0!==s.touchY&&((s.momentum=15*Math.pow(Math.abs(i.pageY-s.touchStartY)/(e.timeStamp-s.touchStartTime),2))&&(this.menuSc
s.touchY)}}},menuShow:function(t){if((t.dataSM("beforefirstshowfired")||(t.dataSM("beforefirstshowfired",!0),this.$root
e=t.dataSM("parent-a"),i=this.isCollapsible();if((this.opts.keepHighlighted||i)&&e.addClass("highlighted"),i)t.removeCl
s=this.getWidth(t);t.css("max-width",this.opts.subMenusMaxWidth),s>this.getWidth(t)&&t.removeClass("sm-nowrap").css("wi
o=function(){t.css("overflow","")};i?canAnimate&&this.opts.collapsibleShowFunction?this.opts.collapsibleShowFunction.ca
e=this;this.hideTimeout=setTimeout(function(){e.menuHideAll()},t?1:this.opts.hideTimeout)},popupShow:function(t,e){if(
alert('SmartMenus jQuery Error:\n\nIf you want to show this menu via the "popupShow" method, set the
isPopup:true option.'),void
0;if(this.hideTimeout&&(clearTimeout(this.hideTimeout),this.hideTimeout=0),this.$root.dataSM("shown-before",!0),canAnim
i=this,s=function(){i.$root.css("overflow","")};canAnimate&&this.opts.showFunction?this.opts.showFunction.call(this,thi
27:var e=this.activatedItems[0];if(e){this.menuHideAll(),e[0].focus();var
i=e.dataSM("sub");i&&this.menuHide(i)}break;case 32:var
```

```
s=$(t.target);if(s.is("a")&&this.handleItemEvents(s)){var
i=s.dataSM("sub");i&&!i.is(":visible")&&(this.itemClick({currentTarget:t.target}),t.preventDefault())}}},rootOut:functi
e=this;this.hideTimeout=setTimeout(function(){e.menuHideAll()},this.opts.hideTimeout)}},rootOver:function(t){this.handl
window)||"orientationchange"==t.type){var
e=this.isCollapsible();this.wasCollapsible&&e||(this.activatedItems.length&&this.activatedItems[this.activatedItems.len
if(this.$disableOverlay){var
i=this.$root.offset();this.$disableOverlay.css({top:i.top,left:i.left,width:this.$root.outerWidth(),height:this.$root.o
e?this.data(t+"_smartmenus",e):this.data(t+"_smartmenus")},$.fn.removeDataSM=function(t){return
this.removeData(t+"_smartmenus")},$.fn.smartmenus=function(options){if("string"==typeof options){var
args=arguments,method=options;return Array.prototype.shift.call(args),this.each(function(){var
t=$(this).data("smartmenus");t&&t[method]&&t[method].apply(t,args)})}return this.each(function(){var
dataOpts=$(this).data("sm-options")||null;if(dataOpts)try{dataOpts=eval("("+dataOpts+")")}catch(e){dataOpts=null,alert
jQuery init:\nInvalid "data-sm-options" attribute value syntax.')}new
$.SmartMenus(this,$.extend({},$.fn.smartmenus.defaults,options,dataOpts))})},$.fn.smartmenus.defaults={isPopup:!1,mainN
```

8.39 html/menu.js File Reference

8.40 menu.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function initMenu(relPath,searchEnabled,serverSide,searchPage,search) {
  function makeTree(data,relPath) {
    var result='';
    if ('children' in data) {
      result+='<ul>';
      for (var i in data.children) {
        result+='<li><a href="'+relPath+data.children[i].url+'">'+
                                data.children[i].text+'</a>'+
                                makeTree(data.children[i],relPath)+'</li>';
      }
      result+='</ul>';
    }
    return result;
  }

  $('#main-nav').append(makeTree(menudata,relPath));
  $('#main-nav').children(':first').addClass('sm sm-dox').attr('id','main-menu');
  if (searchEnabled) {
    if (serverSide) {
      $('#main-menu').append('<li style="float:right"><div id="MSearchBox"
class="MSearchBoxInactive"><div class="left"><form id="FSearchBox" action="'+relPath+searchPage+'"
method="get"><img id="MSearchSelect" src="'+relPath+'search/mag.svg" alt=""/><input type="text"
id="MSearchField" name="query" value="'+search+'" size="20" accesskey="S"
onfocus="searchBox.OnSearchFieldFocus(true)"
onblur="searchBox.OnSearchFieldFocus(false)"></form></div><div class="right"></div></div></li>');
    } else {
      $('#main-menu').append('<li style="float:right"><div id="MSearchBox"
class="MSearchBoxInactive"><span class="left"><img id="MSearchSelect"
src="'+relPath+'search/mag_sel.svg" onmouseover="return searchBox.OnSearchSelectShow()"
onmouseout="return searchBox.OnSearchSelectHide()" alt=""/><input type="text" id="MSearchField"
value="'+search+'" accesskey="S" onfocus="searchBox.OnSearchFieldFocus(true)"
onblur="searchBox.OnSearchFieldFocus(false)"
onkeyup="searchBox.OnSearchFieldChange(event)"/></span><span class="right"><a id="MSearchClose"
href="javascript:searchBox.CloseResultsWindow()"><img id="MSearchCloseImg" border="0"
src="'+relPath+'search/close.svg" alt=""/></a></span></div></li>');
    }
  }
  $('#main-menu').smartmenus();
}
/* @license-end */
```

8.41 user/html/menu.js File Reference

8.42 menu.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function initMenu(relPath,searchEnabled,serverSide,searchPage,search) {
  function makeTree(data,relPath) {
    var result='';
    if ('children' in data) {
      result+='<ul>';
      for (var i in data.children) {
        result+='<li><a href="'+relPath+data.children[i].url+'">'+
                                data.children[i].text+'</a>'+
                                makeTree(data.children[i],relPath)+'</li>';
      }
      result+='</ul>';
    }
    return result;
  }

  $('#main-nav').append(makeTree(menudata,relPath));
  $('#main-nav').children(':first').addClass('sm sm-dox').attr('id','main-menu');
  if (searchEnabled) {
    if (serverSide) {
      $('#main-menu').append('<li style="float:right"><div id="MSearchBox"
class="MSearchBoxInactive"><div class="left"><form id="FSearchBox" action="'+relPath+searchPage+'"
method="get"><img id="MSearchSelect" src="'+relPath+'search/mag.svg" alt=""/><input type="text"
id="MSearchField" name="query" value="'+search+'" size="20" accesskey="S"
onfocus="searchBox.OnSearchFieldFocus(true)"
onblur="searchBox.OnSearchFieldFocus(false)"></form></div><div class="right"></div></div></li>');
    } else {
      $('#main-menu').append('<li style="float:right"><div id="MSearchBox"
class="MSearchBoxInactive"><span class="left"><img id="MSearchSelect"
src="'+relPath+'search/mag_sel.svg" onmouseover="return searchBox.OnSearchSelectShow()"
onmouseout="return searchBox.OnSearchSelectHide()" alt=""/><input type="text" id="MSearchField"
value="'+search+'" accesskey="S" onfocus="searchBox.OnSearchFieldFocus(true)"
onblur="searchBox.OnSearchFieldFocus(false)"
onkeyup="searchBox.OnSearchFieldChange(event)"/></span><span class="right"><a id="MSearchClose"
href="javascript:searchBox.CloseResultsWindow()"><img id="MSearchCloseImg" border="0"
src="'+relPath+'search/close.svg" alt=""/></a></span></div></li>');
    }
  }
  $('#main-menu').smartmenus();
}
/* @license-end */
```

8.43 html/menudata.js File Reference

8.44 menudata.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

```

```
 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
*/
var menudata={children:[
{text:"Main Page",url:"index.html"}]}
```

8.45 user/html/menudata.js File Reference

8.46 menudata.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
*/
var menudata={children:[
{text:"Main Page",url:"index.html"}]}
```

8.47 html/search/search.js File Reference

8.48 search.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```

```
 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function convertToId(search)
{
  var result = ”;
  for (i=0;i<search.length;i++)
  {
    var c = search.charAt(i);
    var cn = c.charCodeAt(0);
    if (c.match(/[a-z0-9\u0080-\uFFFF]/))
    {
      result+=c;
    }
    else if (cn<16)
    {
      result+="_0"+cn.toString(16);
    }
    else
    {
      result+="_"+cn.toString(16);
    }
  }
  return result;
}

function getXPos(item)
{
  var x = 0;
  if (item.offsetWidth)
  {
    while (item && item!=document.body)
    {
      x   += item.offsetLeft;
      item = item.offsetParent;
    }
  }
  return x;
}

function getYPos(item)
{
  var y = 0;
  if (item.offsetWidth)
  {
     while (item && item!=document.body)
     {
       y   += item.offsetTop;
       item = item.offsetParent;
     }
  }
  return y;
}

/* A class handling everything associated with the search panel.

   Parameters:
   name - The name of the global variable that will be
          storing this instance.  Is needed to be able to set timeouts.
   resultPath - path to use for external files
*/
function SearchBox(name, resultsPath, inFrame, label, extension)
{
  if (!name || !resultsPath) {  alert("Missing parameters to SearchBox."); }
  if (!extension || extension == "") { extension = ".html"; }

  // ---------- Instance variables
  this.name                  = name;
  this.resultsPath           = resultsPath;
  this.keyTimeout            = 0;
  this.keyTimeoutLength      = 500;
  this.closeSelectionTimeout = 300;
  this.lastSearchValue       = "";
  this.lastResultsPage       = "";
  this.hideTimeout           = 0;
  this.searchIndex           = 0;
  this.searchActive          = false;
  this.insideFrame           = inFrame;
  this.searchLabel           = label;
  this.extension             = extension;

  // ----------- DOM Elements

  this.DOMSearchField = function()
  {  return document.getElementById("MSearchField");  }

  this.DOMSearchSelect = function()
  {  return document.getElementById("MSearchSelect");  }
```

```

this.DOMSearchSelectWindow = function()
{  return document.getElementById("MSearchSelectWindow");  }

this.DOMPopupSearchResults = function()
{  return document.getElementById("MSearchResults");  }

this.DOMPopupSearchResultsWindow = function()
{  return document.getElementById("MSearchResultsWindow");  }

this.DOMSearchClose = function()
{  return document.getElementById("MSearchClose"); }

this.DOMSearchBox = function()
{  return document.getElementById("MSearchBox");  }

// ------------ Event Handlers

// Called when focus is added or removed from the search field.
this.OnSearchFieldFocus = function(isActive)
{
  this.Activate(isActive);
}

this.OnSearchSelectShow = function()
{
  var searchSelectWindow = this.DOMSearchSelectWindow();
  var searchField        = this.DOMSearchSelect();

  if (this.insideFrame)
  {
    var left = getXPos(searchField);
    var top  = getYPos(searchField);
    left += searchField.offsetWidth + 6;
    top += searchField.offsetHeight;

    // show search selection popup
    searchSelectWindow.style.display='block';
    left -= searchSelectWindow.offsetWidth;
    searchSelectWindow.style.left =  left + 'px';
    searchSelectWindow.style.top  =  top  + 'px';
  }
  else
  {
    var left = getXPos(searchField);
    var top  = getYPos(searchField);
    top += searchField.offsetHeight;

    // show search selection popup
    searchSelectWindow.style.display='block';
    searchSelectWindow.style.left =  left + 'px';
    searchSelectWindow.style.top  =  top  + 'px';
  }

  // stop selection hide timer
  if (this.hideTimeout)
  {
    clearTimeout(this.hideTimeout);
    this.hideTimeout=0;
  }
  return false; // to avoid "image drag" default event
}

this.OnSearchSelectHide = function()
{
  this.hideTimeout = setTimeout(this.name +".CloseSelectionWindow()",
                                this.closeSelectionTimeout);
}

// Called when the content of the search field is changed.
this.OnSearchFieldChange = function(evt)
{
  if (this.keyTimeout) // kill running timer
  {
    clearTimeout(this.keyTimeout);
    this.keyTimeout = 0;
  }

  var e  = (evt) ? evt : window.event; // for IE
  if (e.keyCode==40 || e.keyCode==13)
  {
    if (e.shiftKey==1)
    {
      this.OnSearchSelectShow();
      var win=this.DOMSearchSelectWindow();
      for (i=0;i<win.childNodes.length;i++)
      {
```

```
          var child = win.childNodes[i]; // get span within a
          if (child.className=='SelectItem')
          {
            child.focus();
            return;
          }
        }
        return;
      }
      else
      {
        window.frames.MSearchResults.postMessage("take_focus", "*");
      }
    }
    else if (e.keyCode==27) // Escape out of the search field
    {
      this.DOMSearchField().blur();
      this.DOMPopupSearchResultsWindow().style.display = 'none';
      this.DOMSearchClose().style.display = 'none';
      this.lastSearchValue = '';
      this.Activate(false);
      return;
    }

    // strip whitespaces
    var searchValue = this.DOMSearchField().value.replace(/ +/g, "");

    if (searchValue != this.lastSearchValue) // search value has changed
    {
      if (searchValue != "") // non-empty search
      {
        // set timer for search update
        this.keyTimeout = setTimeout(this.name + '.Search()',
                                     this.keyTimeoutLength);
      }
      else // empty search field
      {
        this.DOMPopupSearchResultsWindow().style.display = 'none';
        this.DOMSearchClose().style.display = 'none';
        this.lastSearchValue = '';
      }
    }
  }

  this.SelectItemCount = function(id)
  {
    var count=0;
    var win=this.DOMSearchSelectWindow();
    for (i=0;i<win.childNodes.length;i++)
    {
      var child = win.childNodes[i]; // get span within a
      if (child.className=='SelectItem')
      {
        count++;
      }
    }
    return count;
  }

  this.SelectItemSet = function(id)
  {
    var i,j=0;
    var win=this.DOMSearchSelectWindow();
    for (i=0;i<win.childNodes.length;i++)
    {
      var child = win.childNodes[i]; // get span within a
      if (child.className=='SelectItem')
      {
        var node = child.firstChild;
        if (j==id)
        {
          node.innerHTML='&#8226;';
        }
        else
        {
          node.innerHTML=' ';
        }
        j++;
      }
    }
  }

  // Called when an search filter selection is made.
  // set item with index id as the active item
  this.OnSelectItem = function(id)
  {
    this.searchIndex = id;
```

```
    this.SelectItemSet(id);
    var searchValue = this.DOMSearchField().value.replace(/ +/g, "");
    if (searchValue!="" && this.searchActive) // something was found -> do a search
    {
      this.Search();
    }
  }

  this.OnSearchSelectKey = function(evt)
  {
    var e = (evt) ? evt : window.event; // for IE
    if (e.keyCode==40 && this.searchIndex<this.SelectItemCount()) // Down
    {
      this.searchIndex++;
      this.OnSelectItem(this.searchIndex);
    }
    else if (e.keyCode==38 && this.searchIndex>0) // Up
    {
      this.searchIndex--;
      this.OnSelectItem(this.searchIndex);
    }
    else if (e.keyCode==13 || e.keyCode==27)
    {
      this.OnSelectItem(this.searchIndex);
      this.CloseSelectionWindow();
      this.DOMSearchField().focus();
    }
    return false;
  }

  // --------- Actions

  // Closes the results window.
  this.CloseResultsWindow = function()
  {
    this.DOMPopupSearchResultsWindow().style.display = 'none';
    this.DOMSearchClose().style.display = 'none';
    this.Activate(false);
  }

  this.CloseSelectionWindow = function()
  {
    this.DOMSearchSelectWindow().style.display = 'none';
  }

  // Performs a search.
  this.Search = function()
  {
    this.keyTimeout = 0;

    // strip leading whitespace
    var searchValue = this.DOMSearchField().value.replace(/^ +/, "");

    var code = searchValue.toLowerCase().charCodeAt(0);
    var idxChar = searchValue.substr(0, 1).toLowerCase();
    if ( 0xD800 <= code && code <= 0xDBFF && searchValue > 1) // surrogate pair
    {
      idxChar = searchValue.substr(0, 2);
    }

    var resultsPage;
    var resultsPageWithSearch;
    var hasResultsPage;

    var idx = indexSectionsWithContent[this.searchIndex].indexOf(idxChar);
    if (idx!=-1)
    {
       var hexCode=idx.toString(16);
       resultsPage = this.resultsPath + '/' + indexSectionNames[this.searchIndex] + '_' + hexCode +
this.extension;
       resultsPageWithSearch = resultsPage+'?'+escape(searchValue);
       hasResultsPage = true;
    }
    else // nothing available for this search term
    {
       resultsPage = this.resultsPath + '/nomatches' + this.extension;
       resultsPageWithSearch = resultsPage;
       hasResultsPage = false;
    }

    window.frames.MSearchResults.location = resultsPageWithSearch;
    var domPopupSearchResultsWindow = this.DOMPopupSearchResultsWindow();

    if (domPopupSearchResultsWindow.style.display!='block')
    {
       var domSearchBox = this.DOMSearchBox();
       this.DOMSearchClose().style.display = 'inline-block';
```

```
      if (this.insideFrame)
      {
        var domPopupSearchResults = this.DOMPopupSearchResults();
        domPopupSearchResultsWindow.style.position = 'relative';
        domPopupSearchResultsWindow.style.display  = 'block';
        var width = document.body.clientWidth - 8; // the -8 is for IE :-(
        domPopupSearchResultsWindow.style.width    = width + 'px';
        domPopupSearchResults.style.width          = width + 'px';
      }
      else
      {
        var domPopupSearchResults = this.DOMPopupSearchResults();
        var left = getXPos(domSearchBox) + 150; // domSearchBox.offsetWidth;
        var top  = getYPos(domSearchBox) + 20;  // domSearchBox.offsetHeight + 1;
        domPopupSearchResultsWindow.style.display = 'block';
        left -= domPopupSearchResults.offsetWidth;
        domPopupSearchResultsWindow.style.top     = top  + 'px';
        domPopupSearchResultsWindow.style.left    = left + 'px';
      }
    }

    this.lastSearchValue = searchValue;
    this.lastResultsPage = resultsPage;
  }

  // -------- Activation Functions

  // Activates or deactivates the search panel, resetting things to
  // their default values if necessary.
  this.Activate = function(isActive)
  {
    if (isActive || // open it
        this.DOMPopupSearchResultsWindow().style.display == 'block'
       )
    {
      this.DOMSearchBox().className = 'MSearchBoxActive';

      var searchField = this.DOMSearchField();

      if (searchField.value == this.searchLabel) // clear "Search" term upon entry
      {
        searchField.value = '';
        this.searchActive = true;
      }
    }
    else if (!isActive) // directly remove the panel
    {
      this.DOMSearchBox().className = 'MSearchBoxInactive';
      this.DOMSearchField().value   = this.searchLabel;
      this.searchActive             = false;
      this.lastSearchValue          = ''
      this.lastResultsPage          = '';
    }
  }
}

// ----------------------------------------------------------------------

// The class that handles everything on the search results page.
function SearchResults(name)
{
    // The number of matches from the last run of <Search()>.
    this.lastMatchCount = 0;
    this.lastKey = 0;
    this.repeatOn = false;

    // Toggles the visibility of the passed element ID.
    this.FindChildElement = function(id)
    {
      var parentElement = document.getElementById(id);
      var element = parentElement.firstChild;

      while (element && element!=parentElement)
      {
        if (element.nodeName.toLowerCase() == 'div' && element.className == 'SRChildren')
        {
          return element;
        }

        if (element.nodeName.toLowerCase() == 'div' && element.hasChildNodes())
        {
           element = element.firstChild;
        }
        else if (element.nextSibling)
        {
           element = element.nextSibling;
        }
```

```
        else
        {
          do
          {
            element = element.parentNode;
          }
          while (element && element!=parentElement && !element.nextSibling);

          if (element && element!=parentElement)
          {
            element = element.nextSibling;
          }
        }
      }
    }

    this.Toggle = function(id)
    {
      var element = this.FindChildElement(id);
      if (element)
      {
        if (element.style.display == 'block')
        {
          element.style.display = 'none';
        }
        else
        {
          element.style.display = 'block';
        }
      }
    }

    // Searches for the passed string.  If there is no parameter,
    // it takes it from the URL query.
    //
    // Always returns true, since other documents may try to call it
    // and that may or may not be possible.
    this.Search = function(search)
    {
      if (!search) // get search word from URL
      {
        search = window.location.search;
        search = search.substring(1);  // Remove the leading '?'
        search = unescape(search);
      }

      search = search.replace(/^ +/, ""); // strip leading spaces
      search = search.replace(/ +$/, ""); // strip trailing spaces
      search = search.toLowerCase();
      search = convertToId(search);

      var resultRows = document.getElementsByTagName("div");
      var matches = 0;

      var i = 0;
      while (i < resultRows.length)
      {
        var row = resultRows.item(i);
        if (row.className == "SRResult")
        {
          var rowMatchName = row.id.toLowerCase();
          rowMatchName = rowMatchName.replace(/^sr\d*_/, ''); // strip 'sr123_'

          if (search.length<=rowMatchName.length &&
             rowMatchName.substr(0, search.length)==search)
          {
            row.style.display = 'block';
            matches++;
          }
          else
          {
            row.style.display = 'none';
          }
        }
        i++;
      }
      document.getElementById("Searching").style.display='none';
      if (matches == 0) // no results
      {
        document.getElementById("NoMatches").style.display='block';
      }
      else // at least one result
      {
        document.getElementById("NoMatches").style.display='none';
      }
      this.lastMatchCount = matches;
      return true;
```

```
    }

    // return the first item with index index or higher that is visible
    this.NavNext = function(index)
    {
      var focusItem;
      while (1)
      {
        var focusName = 'Item'+index;
        focusItem = document.getElementById(focusName);
        if (focusItem && focusItem.parentNode.parentNode.style.display=='block')
        {
          break;
        }
        else if (!focusItem) // last element
        {
          break;
        }
        focusItem=null;
        index++;
      }
      return focusItem;
    }

    this.NavPrev = function(index)
    {
      var focusItem;
      while (1)
      {
        var focusName = 'Item'+index;
        focusItem = document.getElementById(focusName);
        if (focusItem && focusItem.parentNode.parentNode.style.display=='block')
        {
          break;
        }
        else if (!focusItem) // last element
        {
          break;
        }
        focusItem=null;
        index--;
      }
      return focusItem;
    }

    this.ProcessKeys = function(e)
    {
      if (e.type == "keydown")
      {
        this.repeatOn = false;
        this.lastKey = e.keyCode;
      }
      else if (e.type == "keypress")
      {
        if (!this.repeatOn)
        {
          if (this.lastKey) this.repeatOn = true;
          return false; // ignore first keypress after keydown
        }
      }
      else if (e.type == "keyup")
      {
        this.lastKey = 0;
        this.repeatOn = false;
      }
      return this.lastKey!=0;
    }

    this.Nav = function(evt,itemIndex)
    {
      var e  = (evt) ? evt : window.event; // for IE
      if (e.keyCode==13) return true;
      if (!this.ProcessKeys(e)) return false;

      if (this.lastKey==38) // Up
      {
        var newIndex = itemIndex-1;
        var focusItem = this.NavPrev(newIndex);
        if (focusItem)
        {
          var child = this.FindChildElement(focusItem.parentNode.parentNode.id);
          if (child && child.style.display == 'block') // children visible
          {
            var n=0;
            var tmpElem;
            while (1) // search for last child
            {
```

```
              tmpElem = document.getElementById('Item'+newIndex+'_c'+n);
              if (tmpElem)
              {
                focusItem = tmpElem;
              }
              else // found it!
              {
                break;
              }
              n++;
            }
          }
        }
        if (focusItem)
        {
          focusItem.focus();
        }
        else // return focus to search field
        {
           parent.document.getElementById("MSearchField").focus();
        }
      }
      else if (this.lastKey==40) // Down
      {
        var newIndex = itemIndex+1;
        var focusItem;
        var item = document.getElementById('Item'+itemIndex);
        var elem = this.FindChildElement(item.parentNode.parentNode.id);
        if (elem && elem.style.display == 'block') // children visible
        {
          focusItem = document.getElementById('Item'+itemIndex+'_c0');
        }
        if (!focusItem) focusItem = this.NavNext(newIndex);
        if (focusItem)  focusItem.focus();
      }
      else if (this.lastKey==39) // Right
      {
        var item = document.getElementById('Item'+itemIndex);
        var elem = this.FindChildElement(item.parentNode.parentNode.id);
        if (elem) elem.style.display = 'block';
      }
      else if (this.lastKey==37) // Left
      {
        var item = document.getElementById('Item'+itemIndex);
        var elem = this.FindChildElement(item.parentNode.parentNode.id);
        if (elem) elem.style.display = 'none';
      }
      else if (this.lastKey==27) // Escape
      {
        parent.searchBox.CloseResultsWindow();
        parent.document.getElementById("MSearchField").focus();
      }
      else if (this.lastKey==13) // Enter
      {
        return true;
      }
      return false;
    }

    this.NavChild = function(evt,itemIndex,childIndex)
    {
      var e  = (evt) ? evt : window.event; // for IE
      if (e.keyCode==13) return true;
      if (!this.ProcessKeys(e)) return false;

      if (this.lastKey==38) // Up
      {
        if (childIndex>0)
        {
          var newIndex = childIndex-1;
          document.getElementById('Item'+itemIndex+'_c'+newIndex).focus();
        }
        else // already at first child, jump to parent
        {
          document.getElementById('Item'+itemIndex).focus();
        }
      }
      else if (this.lastKey==40) // Down
      {
        var newIndex = childIndex+1;
        var elem = document.getElementById('Item'+itemIndex+'_c'+newIndex);
        if (!elem) // last child, jump to parent next parent
        {
          elem = this.NavNext(itemIndex+1);
        }
        if (elem)
        {
```

```
          elem.focus();
        }
      }
      else if (this.lastKey==27) // Escape
      {
        parent.searchBox.CloseResultsWindow();
        parent.document.getElementById("MSearchField").focus();
      }
      else if (this.lastKey==13) // Enter
      {
        return true;
      }
      return false;
    }
}

function setKeyActions(elem,action)
{
  elem.setAttribute('onkeydown',action);
  elem.setAttribute('onkeypress',action);
  elem.setAttribute('onkeyup',action);
}

function setClassAttr(elem,attr)
{
  elem.setAttribute('class',attr);
  elem.setAttribute('className',attr);
}

function createResults()
{
  var results = document.getElementById("SRResults");
  for (var e=0; e<searchData.length; e++)
  {
    var id = searchData[e][0];
    var srResult = document.createElement('div');
    srResult.setAttribute('id','SR_'+id);
    setClassAttr(srResult,'SRResult');
    var srEntry = document.createElement('div');
    setClassAttr(srEntry,'SREntry');
    var srLink = document.createElement('a');
    srLink.setAttribute('id','Item'+e);
    setKeyActions(srLink,'return searchResults.Nav(event,'+e+')');
    setClassAttr(srLink,'SRSymbol');
    srLink.innerHTML = searchData[e][1][0];
    srEntry.appendChild(srLink);
    if (searchData[e][1].length==2) // single result
    {
      srLink.setAttribute('href',searchData[e][1][1][0]);
      if (searchData[e][1][1][1])
      {
       srLink.setAttribute('target','_parent');
      }
      var srScope = document.createElement('span');
      setClassAttr(srScope,'SRScope');
      srScope.innerHTML = searchData[e][1][1][2];
      srEntry.appendChild(srScope);
    }
    else // multiple results
    {
      srLink.setAttribute('href','javascript:searchResults.Toggle("SR_'+id+'")');
      var srChildren = document.createElement('div');
      setClassAttr(srChildren,'SRChildren');
      for (var c=0; c<searchData[e][1].length-1; c++)
      {
        var srChild = document.createElement('a');
        srChild.setAttribute('id','Item'+e+'_c'+c);
        setKeyActions(srChild,'return searchResults.NavChild(event,'+e+','+c+')');
        setClassAttr(srChild,'SRScope');
        srChild.setAttribute('href',searchData[e][1][c+1][0]);
        if (searchData[e][1][c+1][1])
        {
         srChild.setAttribute('target','_parent');
        }
        srChild.innerHTML = searchData[e][1][c+1][2];
        srChildren.appendChild(srChild);
      }
      srEntry.appendChild(srChildren);
    }
    srResult.appendChild(srEntry);
    results.appendChild(srResult);
  }
}

function init_search()
{
  var results = document.getElementById("MSearchSelectWindow");
```

```
  for (var key in indexSectionLabels)
  {
    var link = document.createElement('a');
    link.setAttribute('class','SelectItem');
    link.setAttribute('onclick','searchBox.OnSelectItem('+key+')');
    link.href='javascript:void(0)';
    link.innerHTML='<span class="SelectionMark"> </span>'+indexSectionLabels[key];
    results.appendChild(link);
  }
  searchBox.OnSelectItem(0);
}
/* @license-end */
```

8.49 user/html/search/search.js File Reference

8.50 search.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function convertToId(search)
{
  var result = '';
  for (i=0;i<search.length;i++)
  {
    var c = search.charAt(i);
    var cn = c.charCodeAt(0);
    if (c.match(/[a-z0-9\u0080-\uFFFF]/))
    {
      result+=c;
    }
    else if (cn<16)
    {
      result+="_0"+cn.toString(16);
    }
    else
    {
      result+="_"+cn.toString(16);
    }
  }
  return result;
}

function getXPos(item)
{
  var x = 0;
  if (item.offsetWidth)
  {
    while (item && item!=document.body)
    {
      x   += item.offsetLeft;
      item = item.offsetParent;
    }
  }
  return x;
}

function getYPos(item)
{
  var y = 0;
  if (item.offsetWidth)
```

```
  {
     while (item && item!=document.body)
     {
       y   += item.offsetTop;
       item = item.offsetParent;
     }
  }
  return y;
}

/* A class handling everything associated with the search panel.

   Parameters:
   name - The name of the global variable that will be
          storing this instance.  Is needed to be able to set timeouts.
   resultPath - path to use for external files
*/
function SearchBox(name, resultsPath, inFrame, label, extension)
{
  if (!name || !resultsPath) {  alert("Missing parameters to SearchBox."); }
  if (!extension || extension == "") { extension = ".html"; }

  // ---------- Instance variables
  this.name                   = name;
  this.resultsPath            = resultsPath;
  this.keyTimeout             = 0;
  this.keyTimeoutLength       = 500;
  this.closeSelectionTimeout  = 300;
  this.lastSearchValue        = "";
  this.lastResultsPage        = "";
  this.hideTimeout            = 0;
  this.searchIndex            = 0;
  this.searchActive           = false;
  this.insideFrame            = inFrame;
  this.searchLabel            = label;
  this.extension              = extension;

  // ----------- DOM Elements

  this.DOMSearchField = function()
  {  return document.getElementById("MSearchField");  }

  this.DOMSearchSelect = function()
  {  return document.getElementById("MSearchSelect");  }

  this.DOMSearchSelectWindow = function()
  {  return document.getElementById("MSearchSelectWindow");  }

  this.DOMPopupSearchResults = function()
  {  return document.getElementById("MSearchResults");  }

  this.DOMPopupSearchResultsWindow = function()
  {  return document.getElementById("MSearchResultsWindow");  }

  this.DOMSearchClose = function()
  {  return document.getElementById("MSearchClose"); }

  this.DOMSearchBox = function()
  {  return document.getElementById("MSearchBox");  }

  // ------------ Event Handlers

  // Called when focus is added or removed from the search field.
  this.OnSearchFieldFocus = function(isActive)
  {
    this.Activate(isActive);
  }

  this.OnSearchSelectShow = function()
  {
    var searchSelectWindow = this.DOMSearchSelectWindow();
    var searchField        = this.DOMSearchSelect();

    if (this.insideFrame)
    {
      var left = getXPos(searchField);
      var top  = getYPos(searchField);
      left += searchField.offsetWidth + 6;
      top += searchField.offsetHeight;

      // show search selection popup
      searchSelectWindow.style.display='block';
      left -= searchSelectWindow.offsetWidth;
      searchSelectWindow.style.left =  left + 'px';
      searchSelectWindow.style.top  =  top  + 'px';
    }
    else
```

```
    {
      var left = getXPos(searchField);
      var top  = getYPos(searchField);
      top += searchField.offsetHeight;

      // show search selection popup
      searchSelectWindow.style.display='block';
      searchSelectWindow.style.left =  left + 'px';
      searchSelectWindow.style.top  =  top  + 'px';
    }

    // stop selection hide timer
    if (this.hideTimeout)
    {
      clearTimeout(this.hideTimeout);
      this.hideTimeout=0;
    }
    return false; // to avoid "image drag" default event
  }

  this.OnSearchSelectHide = function()
  {
    this.hideTimeout = setTimeout(this.name +".CloseSelectionWindow()",
                                  this.closeSelectionTimeout);
  }

  // Called when the content of the search field is changed.
  this.OnSearchFieldChange = function(evt)
  {
    if (this.keyTimeout) // kill running timer
    {
      clearTimeout(this.keyTimeout);
      this.keyTimeout = 0;
    }

    var e  = (evt) ? evt : window.event; // for IE
    if (e.keyCode==40 || e.keyCode==13)
    {
      if (e.shiftKey==1)
      {
        this.OnSearchSelectShow();
        var win=this.DOMSearchSelectWindow();
        for (i=0;i<win.childNodes.length;i++)
        {
          var child = win.childNodes[i]; // get span within a
          if (child.className=='SelectItem')
          {
            child.focus();
            return;
          }
        }
        return;
      }
      else
      {
        window.frames.MSearchResults.postMessage("take_focus", "*");
      }
    }
    else if (e.keyCode==27) // Escape out of the search field
    {
      this.DOMSearchField().blur();
      this.DOMPopupSearchResultsWindow().style.display = 'none';
      this.DOMSearchClose().style.display = 'none';
      this.lastSearchValue = '';
      this.Activate(false);
      return;
    }

    // strip whitespaces
    var searchValue = this.DOMSearchField().value.replace(/ +/g, "");

    if (searchValue != this.lastSearchValue) // search value has changed
    {
      if (searchValue != "") // non-empty search
      {
        // set timer for search update
        this.keyTimeout = setTimeout(this.name + '.Search()',
                                     this.keyTimeoutLength);
      }
      else // empty search field
      {
        this.DOMPopupSearchResultsWindow().style.display = 'none';
        this.DOMSearchClose().style.display = 'none';
        this.lastSearchValue = '';
      }
    }
  }
```

```

this.SelectItemCount = function(id)
{
  var count=0;
  var win=this.DOMSearchSelectWindow();
  for (i=0;i<win.childNodes.length;i++)
  {
    var child = win.childNodes[i]; // get span within a
    if (child.className=='SelectItem')
    {
      count++;
    }
  }
  return count;
}

this.SelectItemSet = function(id)
{
  var i,j=0;
  var win=this.DOMSearchSelectWindow();
  for (i=0;i<win.childNodes.length;i++)
  {
    var child = win.childNodes[i]; // get span within a
    if (child.className=='SelectItem')
    {
      var node = child.firstChild;
      if (j==id)
      {
        node.innerHTML='&#8226;';
      }
      else
      {
        node.innerHTML=' ';
      }
      j++;
    }
  }
}

// Called when an search filter selection is made.
// set item with index id as the active item
this.OnSelectItem = function(id)
{
  this.searchIndex = id;
  this.SelectItemSet(id);
  var searchValue = this.DOMSearchField().value.replace(/ +/g, "");
  if (searchValue!="" && this.searchActive) // something was found -> do a search
  {
    this.Search();
  }
}

this.OnSearchSelectKey = function(evt)
{
  var e = (evt) ? evt : window.event; // for IE
  if (e.keyCode==40 && this.searchIndex<this.SelectItemCount()) // Down
  {
    this.searchIndex++;
    this.OnSelectItem(this.searchIndex);
  }
  else if (e.keyCode==38 && this.searchIndex>0) // Up
  {
    this.searchIndex--;
    this.OnSelectItem(this.searchIndex);
  }
  else if (e.keyCode==13 || e.keyCode==27)
  {
    this.OnSelectItem(this.searchIndex);
    this.CloseSelectionWindow();
    this.DOMSearchField().focus();
  }
  return false;
}

// --------- Actions

// Closes the results window.
this.CloseResultsWindow = function()
{
  this.DOMPopupSearchResultsWindow().style.display = 'none';
  this.DOMSearchClose().style.display = 'none';
  this.Activate(false);
}

this.CloseSelectionWindow = function()
{
  this.DOMSearchSelectWindow().style.display = 'none';
```

```
  }

  // Performs a search.
  this.Search = function()
  {
    this.keyTimeout = 0;

    // strip leading whitespace
    var searchValue = this.DOMSearchField().value.replace(/^ +/, "");

    var code = searchValue.toLowerCase().charCodeAt(0);
    var idxChar = searchValue.substr(0, 1).toLowerCase();
    if ( 0xD800 <= code && code <= 0xDBFF && searchValue > 1) // surrogate pair
    {
      idxChar = searchValue.substr(0, 2);
    }

    var resultsPage;
    var resultsPageWithSearch;
    var hasResultsPage;

    var idx = indexSectionsWithContent[this.searchIndex].indexOf(idxChar);
    if (idx!=-1)
    {
       var hexCode=idx.toString(16);
       resultsPage = this.resultsPath + '/' + indexSectionNames[this.searchIndex] + '_' + hexCode +
this.extension;
       resultsPageWithSearch = resultsPage+'?'+escape(searchValue);
       hasResultsPage = true;
    }
    else // nothing available for this search term
    {
       resultsPage = this.resultsPath + '/nomatches' + this.extension;
       resultsPageWithSearch = resultsPage;
       hasResultsPage = false;
    }

    window.frames.MSearchResults.location = resultsPageWithSearch;
    var domPopupSearchResultsWindow = this.DOMPopupSearchResultsWindow();

    if (domPopupSearchResultsWindow.style.display!='block')
    {
       var domSearchBox = this.DOMSearchBox();
       this.DOMSearchClose().style.display = 'inline-block';
       if (this.insideFrame)
       {
         var domPopupSearchResults = this.DOMPopupSearchResults();
         domPopupSearchResultsWindow.style.position = 'relative';
         domPopupSearchResultsWindow.style.display  = 'block';
         var width = document.body.clientWidth - 8; // the -8 is for IE :-(
         domPopupSearchResultsWindow.style.width    = width + 'px';
         domPopupSearchResults.style.width          = width + 'px';
       }
       else
       {
         var domPopupSearchResults = this.DOMPopupSearchResults();
         var left = getXPos(domSearchBox) + 150; // domSearchBox.offsetWidth;
         var top  = getYPos(domSearchBox) + 20;  // domSearchBox.offsetHeight + 1;
         domPopupSearchResultsWindow.style.display = 'block';
         left -= domPopupSearchResults.offsetWidth;
         domPopupSearchResultsWindow.style.top     = top  + 'px';
         domPopupSearchResultsWindow.style.left    = left + 'px';
       }
    }

    this.lastSearchValue = searchValue;
    this.lastResultsPage = resultsPage;
  }

  // -------- Activation Functions

  // Activates or deactivates the search panel, resetting things to
  // their default values if necessary.
  this.Activate = function(isActive)
  {
    if (isActive || // open it
        this.DOMPopupSearchResultsWindow().style.display == 'block'
       )
    {
      this.DOMSearchBox().className = 'MSearchBoxActive';

      var searchField = this.DOMSearchField();

      if (searchField.value == this.searchLabel) // clear "Search" term upon entry
      {
        searchField.value = '';
        this.searchActive = true;
```

```
      }
    }
    else if (!isActive) // directly remove the panel
    {
      this.DOMSearchBox().className = 'MSearchBoxInactive';
      this.DOMSearchField().value   = this.searchLabel;
      this.searchActive             = false;
      this.lastSearchValue          = ''
      this.lastResultsPage          = '';
    }
  }
}

// ----------------------------------------------------------------------

// The class that handles everything on the search results page.
function SearchResults(name)
{
    // The number of matches from the last run of <Search()>.
    this.lastMatchCount = 0;
    this.lastKey = 0;
    this.repeatOn = false;

    // Toggles the visibility of the passed element ID.
    this.FindChildElement = function(id)
    {
      var parentElement = document.getElementById(id);
      var element = parentElement.firstChild;

      while (element && element!=parentElement)
      {
        if (element.nodeName.toLowerCase() == 'div' && element.className == 'SRChildren')
        {
          return element;
        }

        if (element.nodeName.toLowerCase() == 'div' && element.hasChildNodes())
        {
           element = element.firstChild;
        }
        else if (element.nextSibling)
        {
           element = element.nextSibling;
        }
        else
        {
          do
          {
            element = element.parentNode;
          }
          while (element && element!=parentElement && !element.nextSibling);

          if (element && element!=parentElement)
          {
            element = element.nextSibling;
          }
        }
      }
    }

    this.Toggle = function(id)
    {
      var element = this.FindChildElement(id);
      if (element)
      {
        if (element.style.display == 'block')
        {
          element.style.display = 'none';
        }
        else
        {
          element.style.display = 'block';
        }
      }
    }

    // Searches for the passed string.  If there is no parameter,
    // it takes it from the URL query.
    //
    // Always returns true, since other documents may try to call it
    // and that may or may not be possible.
    this.Search = function(search)
    {
      if (!search) // get search word from URL
      {
        search = window.location.search;
        search = search.substring(1);  // Remove the leading '?'
```

```
        search = unescape(search);
      }

      search = search.replace(/^ +/, ""); // strip leading spaces
      search = search.replace(/ +$/, ""); // strip trailing spaces
      search = search.toLowerCase();
      search = convertToId(search);

      var resultRows = document.getElementsByTagName("div");
      var matches = 0;

      var i = 0;
      while (i < resultRows.length)
      {
        var row = resultRows.item(i);
        if (row.className == "SRResult")
        {
          var rowMatchName = row.id.toLowerCase();
          rowMatchName = rowMatchName.replace(/^sr\d*_/, ”); // strip ’sr123_’

          if (search.length<=rowMatchName.length &&
             rowMatchName.substr(0, search.length)==search)
          {
            row.style.display = ’block’;
            matches++;
          }
          else
          {
            row.style.display = ’none’;
          }
        }
        i++;
      }
      document.getElementById("Searching").style.display=’none’;
      if (matches == 0) // no results
      {
        document.getElementById("NoMatches").style.display=’block’;
      }
      else // at least one result
      {
        document.getElementById("NoMatches").style.display=’none’;
      }
      this.lastMatchCount = matches;
      return true;
    }

    // return the first item with index index or higher that is visible
    this.NavNext = function(index)
    {
      var focusItem;
      while (1)
      {
        var focusName = ’Item’+index;
        focusItem = document.getElementById(focusName);
        if (focusItem && focusItem.parentNode.parentNode.style.display==’block’)
        {
          break;
        }
        else if (!focusItem) // last element
        {
          break;
        }
        focusItem=null;
        index++;
      }
      return focusItem;
    }

    this.NavPrev = function(index)
    {
      var focusItem;
      while (1)
      {
        var focusName = ’Item’+index;
        focusItem = document.getElementById(focusName);
        if (focusItem && focusItem.parentNode.parentNode.style.display==’block’)
        {
          break;
        }
        else if (!focusItem) // last element
        {
          break;
        }
        focusItem=null;
        index--;
      }
      return focusItem;
```

```
    }

    this.ProcessKeys = function(e)
    {
      if (e.type == "keydown")
      {
        this.repeatOn = false;
        this.lastKey = e.keyCode;
      }
      else if (e.type == "keypress")
      {
        if (!this.repeatOn)
        {
          if (this.lastKey) this.repeatOn = true;
          return false; // ignore first keypress after keydown
        }
      }
      else if (e.type == "keyup")
      {
        this.lastKey = 0;
        this.repeatOn = false;
      }
      return this.lastKey!=0;
    }

    this.Nav = function(evt,itemIndex)
    {
      var e  = (evt) ? evt : window.event; // for IE
      if (e.keyCode==13) return true;
      if (!this.ProcessKeys(e)) return false;

      if (this.lastKey==38) // Up
      {
        var newIndex = itemIndex-1;
        var focusItem = this.NavPrev(newIndex);
        if (focusItem)
        {
          var child = this.FindChildElement(focusItem.parentNode.parentNode.id);
          if (child && child.style.display == 'block') // children visible
          {
            var n=0;
            var tmpElem;
            while (1) // search for last child
            {
              tmpElem = document.getElementById('Item'+newIndex+'_c'+n);
              if (tmpElem)
              {
                focusItem = tmpElem;
              }
              else // found it!
              {
                break;
              }
              n++;
            }
          }
        }
        if (focusItem)
        {
          focusItem.focus();
        }
        else // return focus to search field
        {
           parent.document.getElementById("MSearchField").focus();
        }
      }
      else if (this.lastKey==40) // Down
      {
        var newIndex = itemIndex+1;
        var focusItem;
        var item = document.getElementById('Item'+itemIndex);
        var elem = this.FindChildElement(item.parentNode.parentNode.id);
        if (elem && elem.style.display == 'block') // children visible
        {
          focusItem = document.getElementById('Item'+itemIndex+'_c0');
        }
        if (!focusItem) focusItem = this.NavNext(newIndex);
        if (focusItem)  focusItem.focus();
      }
      else if (this.lastKey==39) // Right
      {
        var item = document.getElementById('Item'+itemIndex);
        var elem = this.FindChildElement(item.parentNode.parentNode.id);
        if (elem) elem.style.display = 'block';
      }
      else if (this.lastKey==37) // Left
      {
```

```
        var item = document.getElementById('Item'+itemIndex);
        var elem = this.FindChildElement(item.parentNode.parentNode.id);
        if (elem) elem.style.display = 'none';
      }
      else if (this.lastKey==27) // Escape
      {
        parent.searchBox.CloseResultsWindow();
        parent.document.getElementById("MSearchField").focus();
      }
      else if (this.lastKey==13) // Enter
      {
        return true;
      }
      return false;
    }

    this.NavChild = function(evt,itemIndex,childIndex)
    {
      var e  = (evt) ? evt : window.event; // for IE
      if (e.keyCode==13) return true;
      if (!this.ProcessKeys(e)) return false;

      if (this.lastKey==38) // Up
      {
        if (childIndex>0)
        {
          var newIndex = childIndex-1;
          document.getElementById('Item'+itemIndex+'_c'+newIndex).focus();
        }
        else // already at first child, jump to parent
        {
          document.getElementById('Item'+itemIndex).focus();
        }
      }
      else if (this.lastKey==40) // Down
      {
        var newIndex = childIndex+1;
        var elem = document.getElementById('Item'+itemIndex+'_c'+newIndex);
        if (!elem) // last child, jump to parent next parent
        {
          elem = this.NavNext(itemIndex+1);
        }
        if (elem)
        {
          elem.focus();
        }
      }
      else if (this.lastKey==27) // Escape
      {
        parent.searchBox.CloseResultsWindow();
        parent.document.getElementById("MSearchField").focus();
      }
      else if (this.lastKey==13) // Enter
      {
        return true;
      }
      return false;
    }
}

function setKeyActions(elem,action)
{
  elem.setAttribute('onkeydown',action);
  elem.setAttribute('onkeypress',action);
  elem.setAttribute('onkeyup',action);
}

function setClassAttr(elem,attr)
{
  elem.setAttribute('class',attr);
  elem.setAttribute('className',attr);
}

function createResults()
{
  var results = document.getElementById("SRResults");
  for (var e=0; e<searchData.length; e++)
  {
    var id = searchData[e][0];
    var srResult = document.createElement('div');
    srResult.setAttribute('id','SR_'+id);
    setClassAttr(srResult,'SRResult');
    var srEntry = document.createElement('div');
    setClassAttr(srEntry,'SREntry');
    var srLink = document.createElement('a');
    srLink.setAttribute('id','Item'+e);
    setKeyActions(srLink,'return searchResults.Nav(event,'+e+')');
```

```
    setClassAttr(srLink,'SRSymbol');
    srLink.innerHTML = searchData[e][1][0];
    srEntry.appendChild(srLink);
    if (searchData[e][1].length==2) // single result
    {
      srLink.setAttribute('href',searchData[e][1][1][0]);
      if (searchData[e][1][1][1])
      {
       srLink.setAttribute('target','_parent');
      }
      var srScope = document.createElement('span');
      setClassAttr(srScope,'SRScope');
      srScope.innerHTML = searchData[e][1][1][2];
      srEntry.appendChild(srScope);
    }
    else // multiple results
    {
      srLink.setAttribute('href','javascript:searchResults.Toggle("SR_'+id+'")');
      var srChildren = document.createElement('div');
      setClassAttr(srChildren,'SRChildren');
      for (var c=0; c<searchData[e][1].length-1; c++)
      {
        var srChild = document.createElement('a');
        srChild.setAttribute('id','Item'+e+'_c'+c);
        setKeyActions(srChild,'return searchResults.NavChild(event,'+e+','+c+')');
        setClassAttr(srChild,'SRScope');
        srChild.setAttribute('href',searchData[e][1][c+1][0]);
        if (searchData[e][1][c+1][1])
        {
         srChild.setAttribute('target','_parent');
        }
        srChild.innerHTML = searchData[e][1][c+1][2];
        srChildren.appendChild(srChild);
      }
      srEntry.appendChild(srChildren);
    }
    srResult.appendChild(srEntry);
    results.appendChild(srResult);
  }
}

function init_search()
{
  var results = document.getElementById("MSearchSelectWindow");
  for (var key in indexSectionLabels)
  {
    var link = document.createElement('a');
    link.setAttribute('class','SelectItem');
    link.setAttribute('onclick','searchBox.OnSelectItem('+key+')');
    link.href='javascript:void(0)';
    link.innerHTML='<span class="SelectionMark"> </span>'+indexSectionLabels[key];
    results.appendChild(link);
  }
  searchBox.OnSelectItem(0);
}
/* @license-end */
```

8.51 html/search/searchdata.js File Reference

8.52 searchdata.js

```
var indexSectionsWithContent =
{
};

var indexSectionNames =
{
};

var indexSectionLabels =
{
};

```

8.53 user/html/search/searchdata.js File Reference

8.54 searchdata.js

```
var indexSectionsWithContent =
```

```
{
};

var indexSectionNames =
{
};

var indexSectionLabels =
{
};

```

8.55 user/html/navtree.js File Reference

8.56 navtree.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
var navTreeSubIndices = new Array();
var arrowDown = '&#9660;';
var arrowRight = '&#9658;';

function getData(varName)
{
  var i = varName.lastIndexOf('/');
  var n = i>=0 ? varName.substring(i+1) : varName;
  return eval(n.replace(/\-/g,'_'));
}

function stripPath(uri)
{
  return uri.substring(uri.lastIndexOf('/')+1);
}

function stripPath2(uri)
{
  var i = uri.lastIndexOf('/');
  var s = uri.substring(i+1);
  var m = uri.substring(0,i+1).match(/\/d\w\/d\w\w\/$/);
  return m ? uri.substring(i-6) : s;
}

function hashValue()
{
  return $(location).attr('hash').substring(1).replace(/[^\w\-]/g,'');
}

function hashUrl()
{
  return '#'+hashValue();
}

function pathName()
{
  return $(location).attr('pathname').replace(/[^-A-Za-z0-9+&@#/%?=~_|!:,.;\(\)]/g, '');
}

function localStorageSupported()
{
  try {
```

```
    return 'localStorage' in window && window['localStorage'] !== null && window.localStorage.getItem;
  }
  catch(e) {
    return false;
  }
}

function storeLink(link)
{
  if (!$("#nav-sync").hasClass('sync') && localStorageSupported()) {
      window.localStorage.setItem('navpath',link);
  }
}

function deleteLink()
{
  if (localStorageSupported()) {
    window.localStorage.setItem('navpath','');
  }
}

function cachedLink()
{
  if (localStorageSupported()) {
    return window.localStorage.getItem('navpath');
  } else {
    return '';
  }
}

function getScript(scriptName,func,show)
{
  var head = document.getElementsByTagName("head")[0];
  var script = document.createElement('script');
  script.id = scriptName;
  script.type = 'text/javascript';
  script.onload = func;
  script.src = scriptName+'.js';
  head.appendChild(script);
}

function createIndent(o,domNode,node,level)
{
  var level=-1;
  var n = node;
  while (n.parentNode) { level++; n=n.parentNode; }
  if (node.childrenData) {
    var imgNode = document.createElement("span");
    imgNode.className = 'arrow';
    imgNode.style.paddingLeft=(16*level).toString()+'px';
    imgNode.innerHTML=arrowRight;
    node.plus_img = imgNode;
    node.expandToggle = document.createElement("a");
    node.expandToggle.href = "javascript:void(0)";
    node.expandToggle.onclick = function() {
      if (node.expanded) {
        $(node.getChildrenUL()).slideUp("fast");
        node.plus_img.innerHTML=arrowRight;
        node.expanded = false;
      } else {
        expandNode(o, node, false, false);
      }
    }
    node.expandToggle.appendChild(imgNode);
    domNode.appendChild(node.expandToggle);
  } else {
    var span = document.createElement("span");
    span.className = 'arrow';
    span.style.width   = 16*(level+1)+'px';
    span.innerHTML = ' ';
    domNode.appendChild(span);
  }
}

var animationInProgress = false;

function gotoAnchor(anchor,aname,updateLocation)
{
  var pos, docContent = $('#doc-content');
  var ancParent = $(anchor.parent());
  if (ancParent.hasClass('memItemLeft') ||
      ancParent.hasClass('memtitle') ||
      ancParent.hasClass('fieldname') ||
      ancParent.hasClass('fieldtype') ||
      ancParent.is(':header'))
  {
    pos = ancParent.position().top;
```

```
  } else if (anchor.position()) {
    pos = anchor.position().top;
  }
  if (pos) {
    var dist = Math.abs(Math.min(
               pos-docContent.offset().top,
               docContent[0].scrollHeight-
               docContent.height()-docContent.scrollTop()));
    animationInProgress=true;
    docContent.animate({
      scrollTop: pos + docContent.scrollTop() - docContent.offset().top
    },Math.max(50,Math.min(500,dist)),function(){
      if (updateLocation) window.location.href=aname;
      animationInProgress=false;
    });
  }
}

function newNode(o, po, text, link, childrenData, lastNode)
{
  var node = new Object();
  node.children = Array();
  node.childrenData = childrenData;
  node.depth = po.depth + 1;
  node.relpath = po.relpath;
  node.isLast = lastNode;

  node.li = document.createElement("li");
  po.getChildrenUL().appendChild(node.li);
  node.parentNode = po;

  node.itemDiv = document.createElement("div");
  node.itemDiv.className = "item";

  node.labelSpan = document.createElement("span");
  node.labelSpan.className = "label";

  createIndent(o,node.itemDiv,node,0);
  node.itemDiv.appendChild(node.labelSpan);
  node.li.appendChild(node.itemDiv);

  var a = document.createElement("a");
  node.labelSpan.appendChild(a);
  node.label = document.createTextNode(text);
  node.expanded = false;
  a.appendChild(node.label);
  if (link) {
    var url;
    if (link.substring(0,1)=='^') {
      url = link.substring(1);
      link = url;
    } else {
      url = node.relpath+link;
    }
    a.className = stripPath(link.replace('#',':'));
    if (link.indexOf('#')!=-1) {
      var aname = '#'+link.split('#')[1];
      var srcPage = stripPath(pathName());
      var targetPage = stripPath(link.split('#')[0]);
      a.href = srcPage!=targetPage ? url : "javascript:void(0)";
      a.onclick = function(){
        storeLink(link);
        if (!$(a).parent().parent().hasClass('selected'))
        {
          $('.item').removeClass('selected');
          $('.item').removeAttr('id');
          $(a).parent().parent().addClass('selected');
          $(a).parent().parent().attr('id','selected');
        }
        var anchor = $(aname);
        gotoAnchor(anchor,aname,true);
      };
    } else {
      a.href = url;
      a.onclick = function() { storeLink(link); }
    }
  } else {
    if (childrenData != null)
    {
      a.className = "nolink";
      a.href = "javascript:void(0)";
      a.onclick = node.expandToggle.onclick;
    }
  }

  node.childrenUL = null;
  node.getChildrenUL = function() {
```

```
    if (!node.childrenUL) {
      node.childrenUL = document.createElement("ul");
      node.childrenUL.className = "children_ul";
      node.childrenUL.style.display = "none";
      node.li.appendChild(node.childrenUL);
    }
    return node.childrenUL;
  };

  return node;
}

function showRoot()
{
  var headerHeight = $("#top").height();
  var footerHeight = $("#nav-path").height();
  var windowHeight = $(window).height() - headerHeight - footerHeight;
  (function (){ // retry until we can scroll to the selected item
    try {
      var navtree=$('#nav-tree');
      navtree.scrollTo('#selected',100,{offset:-windowHeight/2});
    } catch (err) {
      setTimeout(arguments.callee, 0);
    }
  })();
}

function expandNode(o, node, imm, showRoot)
{
  if (node.childrenData && !node.expanded) {
    if (typeof(node.childrenData)==='string') {
      var varName    = node.childrenData;
      getScript(node.relpath+varName,function(){
        node.childrenData = getData(varName);
        expandNode(o, node, imm, showRoot);
      }, showRoot);
    } else {
      if (!node.childrenVisited) {
        getNode(o, node);
      }
      $(node.getChildrenUL()).slideDown("fast");
      node.plus_img.innerHTML = arrowDown;
      node.expanded = true;
    }
  }
}

function glowEffect(n,duration)
{
  n.addClass('glow').delay(duration).queue(function(next){
    $(this).removeClass('glow');next();
  });
}

function highlightAnchor()
{
  var aname = hashUrl();
  var anchor = $(aname);
  if (anchor.parent().attr('class')=='memItemLeft'){
    var rows = $('.memberdecls tr[class$="'+hashValue()+'"]');
    glowEffect(rows.children(),300); // member without details
  } else if (anchor.parent().attr('class')=='fieldname'){
    glowEffect(anchor.parent().parent(),1000); // enum value
  } else if (anchor.parent().attr('class')=='fieldtype'){
    glowEffect(anchor.parent().parent(),1000); // struct field
  } else if (anchor.parent().is(":header")) {
    glowEffect(anchor.parent(),1000); // section header
  } else {
    glowEffect(anchor.next(),1000); // normal member
  }
}

function selectAndHighlight(hash,n)
{
  var a;
  if (hash) {
    var link=stripPath(pathName())+':'+hash.substring(1);
    a=$('.item a[class$="'+link+'"]');
  }
  if (a && a.length) {
    a.parent().parent().addClass('selected');
    a.parent().parent().attr('id','selected');
    highlightAnchor();
  } else if (n) {
    $(n.itemDiv).addClass('selected');
    $(n.itemDiv).attr('id','selected');
  }
```

```
  if ($('#nav-tree-contents .item:first').hasClass('selected')) {
    $('#nav-sync').css('top','30px');
  } else {
    $('#nav-sync').css('top','5px');
  }
  showRoot();
}

function showNode(o, node, index, hash)
{
  if (node && node.childrenData) {
    if (typeof(node.childrenData)==='string') {
      var varName    = node.childrenData;
      getScript(node.relpath+varName,function(){
        node.childrenData = getData(varName);
        showNode(o,node,index,hash);
      },true);
    } else {
      if (!node.childrenVisited) {
        getNode(o, node);
      }
      $(node.getChildrenUL()).css({'display':'block'});
      node.plus_img.innerHTML = arrowDown;
      node.expanded = true;
      var n = node.children[o.breadcrumbs[index]];
      if (index+1<o.breadcrumbs.length) {
        showNode(o,n,index+1,hash);
      } else {
        if (typeof(n.childrenData)==='string') {
          var varName = n.childrenData;
          getScript(n.relpath+varName,function(){
            n.childrenData = getData(varName);
            node.expanded=false;
            showNode(o,node,index,hash); // retry with child node expanded
          },true);
        } else {
          var rootBase = stripPath(o.toroot.replace(/\..+$/, ''));
          if (rootBase=="index" || rootBase=="pages" || rootBase=="search") {
            expandNode(o, n, true, true);
          }
          selectAndHighlight(hash,n);
        }
      }
    }
  } else {
    selectAndHighlight(hash);
  }
}

function removeToInsertLater(element) {
  var parentNode = element.parentNode;
  var nextSibling = element.nextSibling;
  parentNode.removeChild(element);
  return function() {
    if (nextSibling) {
      parentNode.insertBefore(element, nextSibling);
    } else {
      parentNode.appendChild(element);
    }
  };
}

function getNode(o, po)
{
  var insertFunction = removeToInsertLater(po.li);
  po.childrenVisited = true;
  var l = po.childrenData.length-1;
  for (var i in po.childrenData) {
    var nodeData = po.childrenData[i];
    po.children[i] = newNode(o, po, nodeData[0], nodeData[1], nodeData[2],
      i==l);
  }
  insertFunction();
}

function gotoNode(o,subIndex,root,hash,relpath)
{
  var nti = navTreeSubIndices[subIndex][root+hash];
  o.breadcrumbs = $.extend(true, [], nti ? nti : navTreeSubIndices[subIndex][root]);
  if (!o.breadcrumbs && root!=NAVTREE[0][1]) { // fallback: show index
    navTo(o,NAVTREE[0][1],"",relpath);
    $('.item').removeClass('selected');
    $('.item').removeAttr('id');
  }
  if (o.breadcrumbs) {
    o.breadcrumbs.unshift(0); // add 0 for root node
    showNode(o, o.node, 0, hash);
```

```
  }
}

function navTo(o,root,hash,relpath)
{
  var link = cachedLink();
  if (link) {
    var parts = link.split('#');
    root = parts[0];
    if (parts.length>1) hash = '#'+parts[1].replace(/[^\w\-]/g,'');
    else hash='';
  }
  if (hash.match(/^#l\d+$/)) {
    var anchor=$('a[name='+hash.substring(1)+']');
    glowEffect(anchor.parent(),1000); // line number
    hash=''; // strip line number anchors
  }
  var url=root+hash;
  var i=-1;
  while (NAVTREEINDEX[i+1]<=url) i++;
  if (i==-1) { i=0; root=NAVTREE[0][1]; } // fallback: show index
  if (navTreeSubIndices[i]) {
    gotoNode(o,i,root,hash,relpath)
  } else {
    getScript(relpath+'navtreeindex'+i,function(){
      navTreeSubIndices[i] = eval('NAVTREEINDEX'+i);
      if (navTreeSubIndices[i]) {
        gotoNode(o,i,root,hash,relpath);
      }
    },true);
  }
}

function showSyncOff(n,relpath)
{
    n.html('<img src="'+relpath+'sync_off.png" title="'+SYNCOFFMSG+'"/>');
}

function showSyncOn(n,relpath)
{
    n.html('<img src="'+relpath+'sync_on.png" title="'+SYNCONMSG+'"/>');
}

function toggleSyncButton(relpath)
{
  var navSync = $('#nav-sync');
  if (navSync.hasClass('sync')) {
    navSync.removeClass('sync');
    showSyncOff(navSync,relpath);
    storeLink(stripPath2(pathName())+hashUrl());
  } else {
    navSync.addClass('sync');
    showSyncOn(navSync,relpath);
    deleteLink();
  }
}

var loadTriggered = false;
var readyTriggered = false;
var loadObject,loadToRoot,loadUrl,loadRelPath;

$(window).on('load',function(){
  if (readyTriggered) { // ready first
    navTo(loadObject,loadToRoot,loadUrl,loadRelPath);
    showRoot();
  }
  loadTriggered=true;
});

function initNavTree(toroot,relpath)
{
  var o = new Object();
  o.toroot = toroot;
  o.node = new Object();
  o.node.li = document.getElementById("nav-tree-contents");
  o.node.childrenData = NAVTREE;
  o.node.children = new Array();
  o.node.childrenUL = document.createElement("ul");
  o.node.getChildrenUL = function() { return o.node.childrenUL; };
  o.node.li.appendChild(o.node.childrenUL);
  o.node.depth = 0;
  o.node.relpath = relpath;
  o.node.expanded = false;
  o.node.isLast = true;
  o.node.plus_img = document.createElement("span");
  o.node.plus_img.className = 'arrow';
  o.node.plus_img.innerHTML = arrowRight;
```

```

  if (localStorageSupported()) {
    var navSync = $('#nav-sync');
    if (cachedLink()) {
      showSyncOff(navSync,relpath);
      navSync.removeClass('sync');
    } else {
      showSyncOn(navSync,relpath);
    }
    navSync.click(function(){ toggleSyncButton(relpath); });
  }

  if (loadTriggered) { // load before ready
    navTo(o,toroot,hashUrl(),relpath);
    showRoot();
  } else { // ready before load
    loadObject  = o;
    loadToRoot  = toroot;
    loadUrl     = hashUrl();
    loadRelPath = relpath;
    readyTriggered=true;
  }

  $(window).bind('hashchange', function(){
     if (window.location.hash && window.location.hash.length>1){
       var a;
       if ($(location).attr('hash')){
         var clslink=stripPath(pathName())+':'+hashValue();
         a=$('.item a[class$="'+clslink.replace(/</g,'\\3c ')+'"]');
       }
       if (a==null || !$(a).parent().parent().hasClass('selected')){
         $('.item').removeClass('selected');
         $('.item').removeAttr('id');
       }
       var link=stripPath2(pathName());
       navTo(o,link,hashUrl(),relpath);
     } else if (!animationInProgress) {
       $('#doc-content').scrollTop(0);
       $('.item').removeClass('selected');
       $('.item').removeAttr('id');
       navTo(o,toroot,hashUrl(),relpath);
     }
  })
}
/* @license-end */
```

8.57 user/html/navtreedata.js File Reference

8.58 navtreedata.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
*/
var NAVTREE =
[
  [ "ReGenESyS - REborned GENeric and Expansible SYstem Simulator", "index.html", ]
];

var NAVTREEINDEX =
[
"index.html"
```

```
];

var SYNCONMSG = 'click to disable panel synchronisation';
var SYNCOFFMSG = 'click to enable panel synchronisation';
```

8.59 user/html/navtreeindex0.js File Reference

8.60 navtreeindex0.js

```
var NAVTREEINDEX0 =
{
"index.html":[],
"pages.html":[]
};
```

8.61 user/html/resize.js File Reference

8.62 resize.js

```
/*
 @licstart  The following is the entire license notice for the JavaScript code in this file.

 The MIT License (MIT)

 Copyright (C) 1997-2020 by Dimitri van Heesch

 Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 and associated documentation files (the "Software"), to deal in the Software without restriction,
 including without limitation the rights to use, copy, modify, merge, publish, distribute,
 sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 furnished to do so, subject to the following conditions:

 The above copyright notice and this permission notice shall be included in all copies or
 substantial portions of the Software.

 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
 BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

 @licend  The above is the entire license notice for the JavaScript code in this file
 */
function initResizable()
{
  var cookie_namespace = 'doxygen';
  var sidenav,navtree,content,header,collapsed,collapsedWidth=0,barWidth=6,desktop_vp=768,titleHeight;

  function readCookie(cookie)
  {
    var myCookie = cookie_namespace+"_"+cookie+"=";
    if (document.cookie) {
      var index = document.cookie.indexOf(myCookie);
      if (index != -1) {
        var valStart = index + myCookie.length;
        var valEnd = document.cookie.indexOf(";", valStart);
        if (valEnd == -1) {
          valEnd = document.cookie.length;
        }
        var val = document.cookie.substring(valStart, valEnd);
        return val;
      }
    }
    return 0;
  }

  function writeCookie(cookie, val, expiration)
  {
    if (val==undefined) return;
    if (expiration == null) {
      var date = new Date();
      date.setTime(date.getTime()+(10*365*24*60*60*1000)); // default expiration is one week
      expiration = date.toGMTString();
    }
    document.cookie = cookie_namespace + "_" + cookie + "=" + val + "; expires=" + expiration+"; path=/";
  }
```

```

  function resizeWidth()
  {
    var windowWidth = $(window).width() + "px";
    var sidenavWidth = $(sidenav).outerWidth();
    content.css({marginLeft:parseInt(sidenavWidth)+"px"});
    writeCookie('width',sidenavWidth-barWidth, null);
  }

  function restoreWidth(navWidth)
  {
    var windowWidth = $(window).width() + "px";
    content.css({marginLeft:parseInt(navWidth)+barWidth+"px"});
    sidenav.css({width:navWidth + "px"});
  }

  function resizeHeight()
  {
    var headerHeight = header.outerHeight();
    var footerHeight = footer.outerHeight();
    var windowHeight = $(window).height() - headerHeight - footerHeight;
    content.css({height:windowHeight + "px"});
    navtree.css({height:windowHeight + "px"});
    sidenav.css({height:windowHeight + "px"});
    var width=$(window).width();
    if (width!=collapsedWidth) {
      if (width<desktop_vp && collapsedWidth>=desktop_vp) {
        if (!collapsed) {
          collapseExpand();
        }
      } else if (width>desktop_vp && collapsedWidth<desktop_vp) {
        if (collapsed) {
          collapseExpand();
        }
      }
      collapsedWidth=width;
    }
    if (location.hash.slice(1)) {
      (document.getElementById(location.hash.slice(1))||document.body).scrollIntoView();
    }
  }

  function collapseExpand()
  {
    if (sidenav.width()>0) {
      restoreWidth(0);
      collapsed=true;
    }
    else {
      var width = readCookie('width');
      if (width>200 && width<$(window).width()) { restoreWidth(width); } else { restoreWidth(200); }
      collapsed=false;
    }
  }

  header  = $("#top");
  sidenav = $("#side-nav");
  content = $("#doc-content");
  navtree = $("#nav-tree");
  footer  = $("#nav-path");
  $(".side-nav-resizable").resizable({resize: function(e, ui) { resizeWidth(); } });
  $(sidenav).resizable({ minWidth: 0 });
  $(window).resize(function() { resizeHeight(); });
  var device = navigator.userAgent.toLowerCase();
  var touch_device = device.match(/(iphone|ipod|ipad|android)/);
  if (touch_device) { /* wider split bar for touch only devices */
    $(sidenav).css({ paddingRight:'20px' });
    $('.ui-resizable-e').css({ width:'20px' });
    $('#nav-sync').css({ right:'34px' });
    barWidth=20;
  }
  var width = readCookie('width');
  if (width) { restoreWidth(width); } else { resizeWidth(); }
  resizeHeight();
  var url = location.href;
  var i=url.indexOf("#");
  if (i>=0) window.location.hash=url.substr(i);
  var _preventDefault = function(evt) { evt.preventDefault(); };
  $("#splitbar").bind("dragstart", _preventDefault).bind("selectstart", _preventDefault);
  $(".ui-resizable-handle").dblclick(collapseExpand);
  $(window).on('load',resizeHeight);
}
/* @license-end */
```

8.63 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/book/Book_Cap02_↩ Example01.cpp File Reference

```
#include "Book_Cap02_Example01.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
```

Include dependency graph for Book_Cap02_Example01.cpp:

8.64 Book_Cap02_Example01.cpp

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.cc to edit this template
 */

/*
 * File:   Book_Cap02_Example01.cpp
 * Author: rlcancian
 *
 * Created on 19 de outubro de 2021, 16:23
 */

#include "Book_Cap02_Example01.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"

Book_Cap02_Example01::Book_Cap02_Example01() {
}

int Book_Cap02_Example01::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    Model* model = genesys->getModels()->newModel();
    EntityType* entityType1 = new EntityType(model, "EntityType1");
    Create* create1 = new Create(model);
    create1->setEntityType(entityType1);
    create1->setTimeBetweenCreationsExpression("NORM(5,2)");
    Delay* delay1 = new Delay(model);
    delay1->setDelayExpression("unif(3,7)");
    Dispose* dispose1 = new Dispose(model);
    create1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(dispose1);
    model->getSimulation()->setReplicationLength(30);
    model->getSimulation()->setNumberOfReplications(100);
    model->getSimulation()->start();
    return 0;
}

```

8.65 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/book/Book_Cap02_Example01.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Book_Cap02_Example01.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Book_Cap02_Example01

8.66 Book_Cap02_Example01.h

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.h to edit this template
 */

```

```
/*
 * File:   Book_Cap02_Example01.h
 * Author: rlcancian
 *
 * Created on 19 de outubro de 2021, 16:23
 */

#ifndef BOOK_CAP02_EXAMPLE01_H
#define BOOK_CAP02_EXAMPLE01_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Book_Cap02_Example01 : public BaseConsoleGenesysApplication {
public:
    Book_Cap02_Example01();
    virtual int main(int argc, char** argv);
};

#endif /* BOOK_CAP02_EXAMPLE01_H */

```

8.67 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_AssignWrite↩ Seizes.cpp File Reference

```
#include "Smart_AssignWriteSeizes.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Write.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/elements/Variable.h"
#include "../../plugins/elements/EntityGroup.h"
#include "../../plugins/elements/Set.h"
```

Include dependency graph for Smart_AssignWriteSeizes.cpp:

8.68 Smart_AssignWriteSeizes.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_AssignWriteSeizes.cpp
 * Author: rlcancian
 *
 * Created on 24 de Setembro de 2019, 20:56
 */

#include "Smart_AssignWriteSeizes.h"

```

```
// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Write.h"

// Model elements
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/elements/Variable.h"
#include "../../plugins/elements/EntityGroup.h"
#include "../../plugins/elements/Set.h"

Smart_AssignWriteSeizes::Smart_AssignWriteSeizes() {
}

int Smart_AssignWriteSeizes::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L2_results); //modelResult); //componentArrival);
    this->setDefaultTraceHandlers(genesys->getTracer());
    this->insertFakePluginsByHand(genesys);

    Model* model = genesys->getModels()->newModel();
    ModelSimulation* sim = model->getSimulation();
    sim->setNumberOfReplications(5);
    sim->setReplicationLength(100);
    EntityType* part = new EntityType(model, "Part");
    Create* create1 = new Create(model);
    create1->setEntityType(part);
    create1->setTimeBetweenCreationsExpression("norm(1.5,0.5)");
    create1->setTimeUnit(Util::TimeUnit::second);
    create1->setEntitiesPerCreation(1);
    Assign* assign1 = new Assign(model);
    assign1->getAssignments()->insert(new Assign::Assignment("varNextIndex", "varNextIndex + 1"));
    assign1->getAssignments()->insert(new Assign::Assignment("index", "varNextIndex"));
    Attribute* attr1 = new Attribute(model, "index");
    Variable* var1 = new Variable(model, "varNextIndex");
    Write* write1 = new Write(model);
    write1->setWriteToType(Write::WriteToType::SCREEN);
    write1->writeElements()->insert(new WritelnText(""));
    write1->writeElements()->insert(new WriteText("Atributo index: "));
    write1->writeElements()->insert(new WritelnExpression("index"));
    write1->writeElements()->insert(new WriteText("Variável nextIndex: "));
    write1->writeElements()->insert(new WritelnExpression("varNextIndex"));
    write1->writeElements()->insert(new WriteText("Quantidade ocupada das máquinas: "));
    write1->writeElements()->insert(new WriteExpression("NR( Machine_1 )"));
    write1->writeElements()->insert(new WriteText(", "));
    write1->writeElements()->insert(new WriteExpression("NR(Machine_2)"));
    write1->writeElements()->insert(new WriteText(", "));
    write1->writeElements()->insert(new WritelnExpression("NR(Machine_3)"));
    write1->writeElements()->insert(new WriteText("Estado das máquinas: "));
    write1->writeElements()->insert(new WriteExpression("STATE(Machine_1)"));
    write1->writeElements()->insert(new WriteText(", "));
    write1->writeElements()->insert(new WriteExpression("STATE(Machine_2)"));
    write1->writeElements()->insert(new WriteText(", "));
    write1->writeElements()->insert(new WritelnExpression("STATE(Machine_3)"));
    write1->writeElements()->insert(new WriteText("Quantidade de máquinas ocupadas no Set: "));
    write1->writeElements()->insert(new WritelnExpression("SETSUM(Machine_Set)"));
    write1->writeElements()->insert(new WriteText("Quantidade de entidades na fila 3: "));
    write1->writeElements()->insert(new WritelnExpression("NQ(Queue_Seize_3)"));
    write1->writeElements()->insert(new WriteText("Somatório do atributo 'index' das entidades na fila 3: "));
    write1->writeElements()->insert(new WritelnExpression("SAQUE(Queue_Seize_3,index)"));
    write1->writeElements()->insert(new WriteText("Valor do atributo 'index' da 2ª entidade na fila 3: "));
    write1->writeElements()->insert(new WritelnExpression("AQUE(Queue_Seize_3,2,index)"));
    write1->writeElements()->insert(new WriteText("Tempo médio das entidades na fila 3: "));
    write1->writeElements()->insert(new WritelnExpression("TAVG(Queue_Seize_3.TimeInQueue)"));
    //
    Resource* machine1 = new Resource(model, "Machine_1");
    machine1->setCapacity(1);
    Resource* machine2 = new Resource(model, "Machine_2");
    machine2->setCapacity(2);
    Resource* machine3 = new Resource(model, "Machine_3");
    machine3->setCapacity(3);
    Set* machSet = new Set(model, "Machine_Set");
    machSet->setSetOfType(Util::TypeOf<Resource>());
```

```
    machSet->getElementSet()->insert(machine1);
    machSet->getElementSet()->insert(machine2);
    machSet->getElementSet()->insert(machine3);
    Decide* decide1 = new Decide(model);
    decide1->getConditions()->insert("NR(Machine_1) < MR(Machine_1)");
    decide1->getConditions()->insert("NR(Machine_2) < MR(Machine_2)");
    Queue* queueSeize1 = new Queue(model, "Queue_Seize_1");
    queueSeize1->setOrderRule(Queue::OrderRule::FIFO);
    Seize* seize1 = new Seize(model);
    seize1->getSeizeRequests()->insert(new SeizableItem(machine1));
    seize1->setQueue(queueSeize1);
    Delay* delay1 = new Delay(model);
    delay1->setDelayExpression("norm(15,1)");
    delay1->setDelayTimeUnit(Util::TimeUnit::second);
    Release* release1 = new Release(model);
    release1->getReleaseRequests()->insert(new SeizableItem(machine1));
    Queue* queueSeize2 = new Queue(model, "Queue_Seize_2");
    queueSeize2->setOrderRule(Queue::OrderRule::FIFO);
    Seize* seize2 = new Seize(model);
    seize2->getSeizeRequests()->insert(new SeizableItem(machine2));
    seize2->setQueue(queueSeize2);
    Delay* delay2 = new Delay(model);
    delay2->setDelayExpression("norm(15,1)");
    delay2->setDelayTimeUnit(Util::TimeUnit::second);
    Release* release2 = new Release(model);
    release2->getReleaseRequests()->insert(new SeizableItem(machine2));
    Queue* queueSeize3 = new Queue(model, "Queue_Seize_3");
    queueSeize3->setOrderRule(Queue::OrderRule::FIFO);
    Seize* seize3 = new Seize(model);
    seize3->getSeizeRequests()->insert(new SeizableItem(machine3));
    seize3->setQueue(queueSeize3);
    Delay* delay3 = new Delay(model);
    delay3->setDelayExpression("norm(15,1)");
    delay3->setDelayTimeUnit(Util::TimeUnit::second);
    Release* release3 = new Release(model);
    release3->getReleaseRequests()->insert(new SeizableItem(machine3));
    Dispose* dispose1 = new Dispose(model);
    //
    create1->getConnections()->insert(assign1);
    assign1->getConnections()->insert(write1);
    write1->getConnections()->insert(decide1);
    decide1->getConnections()->insert(seize1);
    decide1->getConnections()->insert(seize2);
    decide1->getConnections()->insert(seize3);
    seize1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(release1);
    release1->getConnections()->insert(dispose1);
    seize2->getConnections()->insert(delay2);
    delay2->getConnections()->insert(release2);
    release2->getConnections()->insert(dispose1);
    seize3->getConnections()->insert(delay3);
    delay3->getConnections()->insert(release3);
    release3->getConnections()->insert(dispose1);
    //
    model->save("./models/Smart_StatationRouteSequence.txt");
    sim->start();
    return 0;
}
```

8.69 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_AssignWrite↩ Seizes.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_AssignWriteSeizes.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_AssignWriteSeizes

8.70 Smart_AssignWriteSeizes.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_AssignWriteSeizes.h
 * Author: rlcancian
 *
 * Created on 24 de Setembro de 2019, 20:56
 */

#ifndef SMART_ASSIGNWRITE3SEIZES_H
```

```
#define SMART_ASSIGNWRITE3SEIZES_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_AssignWriteSeizes : public BaseConsoleGenesysApplication {
public:
    Smart_AssignWriteSeizes();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_ASSIGNWRITE3SEIZES_H */

```

8.71 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Delay.cpp File Reference

```
#include "Smart_Delay.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_Delay.cpp:

8.72 Smart_Delay.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Delay.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#include "Smart_Delay.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_Delay::Smart_Delay() {
}

int Smart_Delay::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
```

```
    // insert "fake plugins" since plugins based on dynamic loaded library are not implemented yet
    this->insertFakePluginsByHand(genesys);
    // Handle traces and simulation events to output them
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L6_arrival);
    Model* model = genesys->getModels()->newModel();
    // build the simulation model
    // if no ModelInfo is provided, then the model will be simulated once (one replication) and the
 replication length will be 3600 seconds (simulated time)
    model->getSimulation()->setReplicationLength(60);
    // create a (Source)ModelElement of type EntityType, used by a ModelComponent that follows
    EntityType* entityType1 = new EntityType(model, "Type_of_Representative_Entity");
    // create a ModelComponent of type Create, used to insert entities into the model
    Create* create1 = new Create(model);
    create1->setEntityType(entityType1);
    create1->setTimeBetweenCreationsExpression("1.5"); // create one new entity every 1.5 seconds
    // create a ModelComponent of type Delay, used to represent a time delay
    Delay* delay1 = new Delay(model);
    // if nothing else is set, the delay will take 1 second
    // create a (Sink)ModelComponent of type Dispose, used to remove entities from the model
    Dispose* dispose1 = new Dispose(model); // insert the component into the model
    // connect model components to create a "workflow" -- should always start from a
 SourceModelComponent and end at a SinkModelComponent (it will be checked)
    create1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(dispose1);
    // save the model into a text file
    model->save("./models/Smart_Delay.txt");
    // execute the simulation util completed and show the report
    model->getSimulation()->start();
    genesys->~Simulator();
    return 0;
};

```

8.73 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Delay.h File Reference

`#include "../../applications/BaseConsoleGenesysApplication.h"`

Include dependency graph for Smart_Delay.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_Delay

8.74 Smart_Delay.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Delay.h
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#ifndef SMART_DELAY_H
#define SMART_DELAY_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_Delay : public BaseConsoleGenesysApplication {
public:
    Smart_Delay();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_DELAY_H */

```

8.75 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Dummy.cpp File Reference

```
#include "Smart_Dummy.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
```

#include "../../kernel/simulator/EntityType.h"

Include dependency graph for Smart_Dummy.cpp:

8.76 Smart_Dummy.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Dummy.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#include "Smart_Dummy.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_Dummy::Smart_Dummy() {
}

int Smart_Dummy::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    // insert "fake plugins" since plugins based on dynamic loaded library are not implemented yet
    this->insertFakePluginsByHand(genesys);
    // Handle traces and simulation events to output them
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L6_arrival);
    Model* model = genesys->getModels()->newModel();
    //model->load("./models/Smart_CreateDelayDispose.txt");
    //model->getSimulation()->start();
    //return 0;
    //
    // build the simulation model
    // if no ModelInfo is provided, then the model will be simulated once (one replication) and the replication length will be 3600 seconds (simulated time)
    model->getSimulation()->setReplicationLength(60);
    // create a (Source)ModelElement of type EntityType, used by a ModelComponent that follows
    EntityType* entityType1 = new EntityType(model, "Type_of_Representative_Entity");
    // create a ModelComponent of type Create, used to insert entities into the model
    Create* create1 = new Create(model);
    create1->setEntityType(entityType1);
    create1->setTimeBetweenCreationsExpression("1.5"); // create one new entity every 1.5 seconds
    // create a ModelComponent of type Delay, used to represent a time delay
    Delay* delay1 = new Delay(model);
    // if nothing else is set, the delay will take 1 second
    // create a (Sink)ModelComponent of type Dispose, used to remove entities from the model
    Dispose* dispose1 = new Dispose(model); // insert the component into the model
    // connect model components to create a "workflow" -- should always start from a SourceModelComponent and end at a SinkModelComponent (it will be checked)
    create1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(dispose1);
    // save the model into a text file
    model->save("./models/Smart_CreateDelayDispose.txt");
    // execute the simulation util completed and show the report
```

```
    model->getSimulation()->start();
    genesys->~Simulator();
    return 0;
};

```

8.77 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Dummy.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_Dummy.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_Dummy

8.78 Smart_Dummy.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Dummy.h
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#ifndef SMART_DUMMY_H
#define SMART_DUMMY_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_Dummy : public BaseConsoleGenesysApplication {
public:
    Smart_Dummy();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_DUMMY_H */

```

8.79 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_HoldSignal.cpp File Reference

```
#include "Smart_HoldSignal.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Hold.h"
#include "../../plugins/components/Signal.h"
#include "../../plugins/components/Dispose.h"
```

Include dependency graph for Smart_HoldSignal.cpp:

8.80 Smart_HoldSignal.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_HoldSignal.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

```

```
#include "Smart_HoldSignal.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Hold.h"
#include "../../plugins/components/Signal.h"
#include "../../plugins/components/Dispose.h"

// Model elements

Smart_HoldSignal::Smart_HoldSignal() {
}

int Smart_HoldSignal::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    //
    Model* model = genesys->getModels()->newModel();

    //
    model->getSimulation()->setReplicationLength(20);
    model->save("./models/Smart_HoldSignal.txt");
    do {
        model->getSimulation()->step();
        std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (model->getSimulation()->isPaused());
    return 0;
};

```

8.81 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_HoldSignal.h File Reference

`#include "../../applications/BaseConsoleGenesysApplication.h"`

Include dependency graph for Smart_HoldSignal.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_HoldSignal

8.82 Smart_HoldSignal.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_HoldSignal.h
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#ifndef SMART_HOLDSIGNAL_H
#define SMART_HOLDSIGNAL_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_HoldSignal : public BaseConsoleGenesysApplication {
public:
    Smart_HoldSignal();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_HOLDSIGNAL_H */

```

8.83 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ModelInfo↩ ModelSimulation.cpp File Reference

```
#include "Smart_ModelInfoModelSimulation.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
```

```
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_ModelInfoModelSimulation.cpp:

8.84 Smart_ModelInfoModelSimulation.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_ModelInfoModelSimulation.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:59
 */

#include "Smart_ModelInfoModelSimulation.h"
// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_ModelInfoModelSimulation::Smart_ModelInfoModelSimulation() {
}

int Smart_ModelInfoModelSimulation::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L6_arrival);
    this->setDefaultTraceHandlers(genesys->getTracer());
    this->insertFakePluginsByHand(genesys);
    Model* model = genesys->getModels()->newModel();
    // set general info about the model
    ModelInfo* infos = model->getInfos();
    infos->setAnalystName("Your name");
    infos->setProjectTitle("The title of the project");
    infos->setDescription("This simulation model tests one of the most basic models possible.");
    infos->setVersion("1.0");
    //
    Create* create1 = new Create(model);
    create1->setEntityType(new EntityType(model));
    create1->setTimeUnit(Util::TimeUnit::minute);
    Delay* delay1 = new Delay(model);
    delay1->setDelayExpression("NORM(1,0.2)");
    delay1->setDelayTimeUnit(Util::TimeUnit::minute);
    Dispose* dispose1 = new Dispose(model);
    create1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(dispose1);
    // set model simulation
    ModelSimulation* sim = model->getSimulation();
    sim->setReplicationLength(15);
    sim->setReplicationLengthTimeUnit(Util::TimeUnit::minute); // each replication will last 15 minutes (simulated time)
    sim->setNumberOfReplications(2); // replicates the simulation 2 times
    sim->setReplicationReportBaseTimeUnit(Util::TimeUnit::minute);
    sim->setWarmUpPeriod(1);
    sim->setWarmUpPeriodTimeUnit(Util::TimeUnit::minute);
    sim->setPauseOnReplication(true);
    sim->setPauseOnEvent(false);
    sim->setShowReportsAfterReplication(false);
    sim->setShowReportsAfterSimulation(true);
```

```
    sim->setStepByStep(true);
    sim ->setTerminatingCondition("");
    sim->getBreakpointsOnComponent()->insert(delay1);
    sim->getBreakpointsOnTime()->insert(6.5);
    sim->getBreakpointsOnTime()->insert(14.0);
    //
    model->save("./models/Smart_ModelInfoModelSimulation.txt");
    do {
        sim->start();
        std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (sim->isPaused());
    return 0;
};

```

8.85 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ModelInfo↩ ModelSimulation.h File Reference

`#include "../../applications/BaseConsoleGenesysApplication.h"`

Include dependency graph for Smart_ModelInfoModelSimulation.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_ModelInfoModelSimulation

8.86 Smart_ModelInfoModelSimulation.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_ModelInfoModelSimulation.h
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:59
 */

#ifndef SMART_MODELINFOMODELSIMULATION_H
#define SMART_MODELINFOMODELSIMULATION_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_ModelInfoModelSimulation : public BaseConsoleGenesysApplication {
public:
    Smart_ModelInfoModelSimulation();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_MODELINFOMODELSIMULATION_H */

```

8.87 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_OnEvent.cpp File Reference

```
#include "Smart_OnEvent.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_OnEvent.cpp:

8.88 Smart_OnEvent.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_OnEvent.cpp
 * Author: rlcancian
```

```
 *
 * Created on 5 de Janeiro de 2021, 18:00
 */

#include "Smart_OnEvent.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_OnEvent::Smart_OnEvent() {
}

void Smart_OnEvent::onBreakpointHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Breakpoint occured at time " << std::to_string(re->getSimulatedTime()) << std::endl;
}

void Smart_OnEvent::onEntityCreateHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Entity " << re->getEntityCreated()->getName() << " was created." << std::endl;
}

void Smart_OnEvent::onEntityMoveHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Entity " << re->getCurrentEntity()->getName() << " was moved to component" << re->getDestinationComponent()->getName() << std::endl;
}

void Smart_OnEvent::onEntityRemoveHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Entity " << re->getCurrentEvent()->getEntity()->getName() << " was removed." << std::endl;
}

void Smart_OnEvent::onProcessEventHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Processing event " << re->getCurrentEvent()->show() << std::endl;
}

void Smart_OnEvent::onReplicationEndHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Replication " << re->getCurrentReplicationNumber() << " ending." << std::endl;
}

void Smart_OnEvent::onReplicationStartHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Replication " << re->getCurrentReplicationNumber() << " starting." << std::endl;
}

void Smart_OnEvent::onReplicationStepHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Replication step at time " << std::to_string(re->getSimulatedTime()) << std::endl;
}

void Smart_OnEvent::onSimulationStartHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Simulation is starting" << std::endl;
}

void Smart_OnEvent::onSimulationEndHandler(SimulationEvent* re) {
    std::cout << "Event (Handler) " << "Replication " << re->getCurrentReplicationNumber() << " ending." << std::endl;
}

void Smart_OnEvent::onSimulationPausedHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Simulation was paused at time " << std::to_string(re->getSimulatedTime()) << std::endl;
}

void Smart_OnEvent::onSimulationResumeHandler(SimulationEvent* re) {
    std::cout << "(Event Handler) " << "Simulation was resumed." << std::endl;
}

int Smart_OnEvent::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    Model* model = genesys->getModels()->newModel();
    model->getTracer()->setTraceLevel(Util::TraceLevel::L0_noTraces); // no traces!!
    //
    OnEventManager* oem = model->getOnEvents();
    oem->addOnBreakpointHandler(this, &Smart_OnEvent::onBreakpointHandler);
```

```
    oem->addOnEntityCreateHandler(this, &Smart_OnEvent::onEntityCreateHandler);
    oem->addOnEntityMoveHandler(this, &Smart_OnEvent::onEntityMoveHandler);
    oem->addOnEntityRemoveHandler(this, &Smart_OnEvent::onEntityRemoveHandler);
    oem->addOnProcessEventHandler(this, &Smart_OnEvent::onProcessEventHandler);
    oem->addOnReplicationEndHandler(this, &Smart_OnEvent::onReplicationEndHandler);
    oem->addOnReplicationStartHandler(this, &Smart_OnEvent::onReplicationStartHandler);
    oem->addOnReplicationStepHandler(this, &Smart_OnEvent::onReplicationStepHandler);
    oem->addOnSimulationEndHandler(this, &Smart_OnEvent::onSimulationEndHandler);
    oem->addOnSimulationPausedHandler(this, &Smart_OnEvent::onSimulationPausedHandler);
    oem->addOnSimulationResumeHandler(this, &Smart_OnEvent::onSimulationResumeHandler);
    oem->addOnSimulationStartHandler(this, &Smart_OnEvent::onSimulationStartHandler);
    //
    Create* create1 = new Create(model);
    create1->setEntityType(new EntityType(model));
    Delay* delay1 = new Delay(model);
    Dispose* dispose1 = new Dispose(model);
    create1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(dispose1);
    model->save("./models/Smart_OnEvent.txt");
    do {
        model->getSimulation()->step();
        std::cout « "Press ENTER to continue...";
        std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (model->getSimulation()->isPaused());
    genesys->~Simulator();
    return 0;
};

```

8.89 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_OnEvent.h File Reference

`#include "../../applications/BaseConsoleGenesysApplication.h"`

Include dependency graph for Smart_OnEvent.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_OnEvent

8.90 Smart_OnEvent.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_OnEvent.h
 * Author: rlcancian
 *
 * Created on 5 de Janeiro de 2021, 18:00
 */

#ifndef SMART_ONEVENT_H
#define SMART_ONEVENT_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_OnEvent : public BaseConsoleGenesysApplication {
public:
    Smart_OnEvent();
public:
    virtual int main(int argc, char** argv);
public:
    void onBreakpointHandler(SimulationEvent* re);
    void onEntityCreateHandler(SimulationEvent* re);
    void onEntityMoveHandler(SimulationEvent* re);
    void onSimulationStartHandler(SimulationEvent* re);
    void onReplicationStartHandler(SimulationEvent* re);
    void onReplicationStepHandler(SimulationEvent* re);
    void onProcessEventHandler(SimulationEvent* re);
    void onReplicationEndHandler(SimulationEvent* re);
    void onSimulationEndHandler(SimulationEvent* re);
    void onSimulationPausedHandler(SimulationEvent* re);
    void onSimulationResumeHandler(SimulationEvent* re);
    void onEntityRemoveHandler(SimulationEvent* re);
};

#endif /* SMART_ONEVENT_H */

```

8.91 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Parser.cpp File Reference

```
#include "Smart_Parser.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
```

```
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_Parser.cpp:

8.92 Smart_Parser.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Parser.cpp
 * Author: rlcancian
 *
 * Created on 5 de Janeiro de 2022, 07:21
 */

#include "Smart_Parser.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_Parser::Smart_Parser() {
}

int Smart_Parser::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    Model* model = genesys->getModels()->newModel();
    std::vector<std::string> expressions = {
        "-3 + 2*0.25 +0xc/4 -1.34e-3 *(-1*-1)-2^5",
        "sin(0.3) - cos(1.75)",
        "trunc(12.99) + round(12.99) + frac(45.009)",
        "sin(0.3) - cos(1.75)",
        "exp(1) -ln(exp(1)) + log(10) + sqrt(81) - mod(15,8)",
        "not false and (3 nand (0 xor false))",
        "norm(0,1) + expo(10) + unif(2,5) + weib(2,1) + logn(10,1) + erla(2,1) + beta(6,2,1,1) + tria(1,4,5)",
        "disc(0.1,1,0.3,2,0.6,3,1,4)", // not implemented in parser
        "if 0 if 1 0 else 1 else if 1 2 else 3"
    };
    for (unsigned int i = 0; i < expressions.size(); i++) {
        std::cout « "\"" « expressions[i] « "\" = " « model->parseExpression(expressions[i]) « std::endl;
    }
    return 0;
};

```

8.93 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Parser.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_Parser.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_Parser

8.94 Smart_Parser.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
/*
 * File:   Smart_Parser.h
 * Author: rlcancian
 *
 * Created on 5 de Janeiro de 2022, 07:21
 */

#ifndef SMART_PARSER_H
#define SMART_PARSER_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_Parser : public BaseConsoleGenesysApplication {
public:
    Smart_Parser();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_PARSER_H */

```

8.95 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ParserModel↩ Functions.cpp File Reference

```
#include "Smart_ParserModelFunctions.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Write.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_ParserModelFunctions.cpp:

8.96 Smart_ParserModelFunctions.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_ParserModelFunctions.cpp
 * Author: rlcancian
 *
 * Created on 5 de Janeiro de 2022, 07:21
 */

#include "Smart_ParserModelFunctions.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

```

```
// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Write.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"
#include "plugins/components/Seize.h"

Smart_ParserModelFunctions::Smart_ParserModelFunctions() {
}

int Smart_ParserModelFunctions::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L2_results);
    Model* m = genesys->getModels()->newModel();
    Create* c = new Create(m);
    c->setEntityType(new EntityType(m));
    Seize* s = new Seize(m);
    s->setQueueableItem(new QueueableItem(new Queue(m)));
    SeizableItem *si = new SeizableItem(new Resource(m));
    s->getSeizeRequests()->insert(si);
    Delay* de = new Delay(m);
    Write* w = new Write(m);
    Release* r = new Release(m);
    r->getReleaseRequests()->insert(si);
    Dispose* di = new Dispose(m);
    c->getConnections()->insert(s);
    s->getConnections()->insert(de);
    de->getConnections()->insert(w);
    w->getConnections()->insert(r);
    r->getConnections()->insert(di);
    //
    w->writeElements()->insert(new WritelnExpression(""));
    //
    m->getSimulation()->start();
    return 0;
};

```

8.97 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ParserModelFunctions.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_ParserModelFunctions.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_ParserModelFunctions

8.98 Smart_ParserModelFunctions.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_ParserModelFunctions.h
 * Author: rlcancian
 *
 * Created on 5 de Janeiro de 2022, 07:21
 */

#ifndef SMART_PARSERMODELFUNCTIONS_H
```

```
#define SMART_PARSERMODELFUNCTIONS_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_ParserModelFunctions : public BaseConsoleGenesysApplication {
public:
    Smart_ParserModelFunctions();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_PARSERMODELFUNCTIONS_H */

```

8.99 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Process.cpp File Reference

```
#include "Smart_Process.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Process.h"
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_Process.cpp:

8.100 Smart_Process.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Process.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#include "Smart_Process.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Process.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_Process::Smart_Process() {
}

```

```
int Smart_Process::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    Model* model = genesys->getModels()->newModel();
    Create *create = new Create(model);
    create->setEntityType(new EntityType(model, "Client"));
    create->setTimeBetweenCreationsExpression("1");
    Process* process = new Process(model);
    process->getSeizeRequests()->insert(new SeizableItem(new Resource(model)));
    process->setQueueableItem(new QueueableItem(new Queue(model)));
    process->setDelayExpression("unif(0.8,1.2)");
    Dispose* dispose = new Dispose(model);
    create->getConnections()->insert(process);
    process->getConnections()->insert(dispose);
    model->getSimulation()->setReplicationLength(10);
    model->save("./models/Smart_Process.txt");
    do {
        model->getSimulation()->step();
        //      std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (model->getSimulation()->isPaused());
    genesys->~Simulator();
    return 0;
};

```

8.101 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Process.h File Reference

`#include "../../applications/BaseConsoleGenesysApplication.h"`

Include dependency graph for Smart_Process.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_Process

8.102 Smart_Process.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Process.h
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#ifndef SMART_PROCESS_H
#define SMART_PROCESS_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_Process : public BaseConsoleGenesysApplication {
public:
    Smart_Process();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_PROCESS_H */

```

8.103 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Process↩ Set.cpp File Reference

```
#include "Smart_ProcessSet.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Process.h"
#include "../../plugins/components/Dispose.h"
```

```
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_ProcessSet.cpp:

8.104 Smart_ProcessSet.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_ProcessSet.cpp
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#include "Smart_ProcessSet.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Process.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_ProcessSet::Smart_ProcessSet() {
}

int Smart_ProcessSet::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L8_detailed);
    Model* model = genesys->getModels()->newModel();
    Create *create = new Create(model);
    create->setEntityType(new EntityType(model, "Client"));
    create->setTimeBetweenCreationsExpression("1");
    Set* set = new Set(model);
    set->setSetOfType("Resource");
    set->getElementSet()->insert(new Resource(model));
    set->getElementSet()->insert(new Resource(model));
    set->getElementSet()->insert(new Resource(model));
    set->getElementSet()->insert(new Resource(model));
    set->getElementSet()->insert(new Resource(model));
    Process* process = new Process(model);
    process->getSeizeRequests()->insert(new SeizableItem(set)); //, SeizableItem::SeizableType::SET));
    process->setQueueableItem(new QueueableItem(new Queue(model)));
    process->setDelayExpression("unif(0.8,1.2)");
    Dispose* dispose = new Dispose(model);
    create->getConnections()->insert(process);
    process->getConnections()->insert(dispose);
    model->getSimulation()->setReplicationLength(10);
    model->save("./models/Smart_ProcessSet.txt");
    do {
        model->getSimulation()->step();
        //      std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (model->getSimulation()->isPaused());
    genesys->~Simulator();
    return 0;
};

```

8.105 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_ProcessSet.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_ProcessSet.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_ProcessSet

8.106 Smart_ProcessSet.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
```

```
00006
00007 /*
00008  * File:   Smart_ProcessSet.h
00009  * Author: rlcancian
00010  *
00011  * Created on 3 de Setembro de 2019, 18:34
00012  */
00013
00014 #ifndef SMART_PROCESSSET_H
00015 #define SMART_PROCESSSET_H
00016
00017 #include "../../applications/BaseConsoleGenesysApplication.h"
00018
00019 class Smart_ProcessSet : public BaseConsoleGenesysApplication {
00020 public:
00021     Smart_ProcessSet();
00022 public:
00023     virtual int main(int argc, char** argv);
00024 };
00025
00026 #endif /* SMART_PROCESSSET_H */
00027
```

8.107 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Route↩ Station.cpp File Reference

```
#include "Smart_RouteStation.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Route.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
```

Include dependency graph for Smart_RouteStation.cpp:

8.108 Smart_RouteStation.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Smart_RouteStation.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 3 de Setembro de 2019, 18:34
00012  */
00013
00014 #include "Smart_RouteStation.h"
00015
00016 // you have to included need libs
00017
00018 // GEnSyS Simulator
00019 #include "../../kernel/simulator/Simulator.h"
00020
```

```
// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Route.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"

// Model elements
//#include "../../kernel/simulator/EntityType.h"

Smart_RouteStation::Smart_RouteStation() {
}

int Smart_RouteStation::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    Model* model = genesys->getModels()->newModel();
    Create* create1 = new Create(model);
    create1->setEntityType(new EntityType(model));
    create1->setTimeBetweenCreationsExpression("4.0");
    Station* station1 = new Station(model);
    Route* route1 = new Route(model);
    route1->setStation(station1);
    route1->setRouteTimeExpression("0.1");
    //
    Enter* enter1 = new Enter(model);
    enter1->setStation(station1);
    Delay* delay1 = new Delay(model);
    Station* station2 = new Station(model);
    Route* route2 = new Route(model);
    route2->setStation(station2);
    route2->setRouteTimeExpression("0.2");
    //
    Enter* enter2 = new Enter(model);
    enter2->setStation(station2);
    Delay* delay2 = new Delay(model);
    Station* station3 = new Station(model);
    Route* route3 = new Route(model);
    route3->setStation(station3);
    route3->setRouteTimeExpression("0.3");
    //
    Enter* enter3 = new Enter(model);
    enter3->setStation(station3);
    Dispose* dispose1 = new Dispose(model);
    //
    create1->getConnections()->insert(route1);
    enter1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(route2);
    enter2->getConnections()->insert(delay2);
    delay2->getConnections()->insert(route3);
    enter3->getConnections()->insert(dispose1);
    //
    model->getSimulation()->setReplicationLength(20);
    model->save("./models/Smart_RouteStation.txt");
    /*
    do {
        model->getSimulation()->step();
        std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (model->getSimulation()->isPaused());
     */
    model->getSimulation()->start();
    return 0;
};

```

8.109 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Route↩ Station.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_RouteStation.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_RouteStation

8.110 Smart_RouteStation.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_RouteStation.h
 * Author: rlcancian
 *
 * Created on 3 de Setembro de 2019, 18:34
 */

#ifndef SMART_ROUTESTATION_H
```

```
#define SMART_ROUTESTATION_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_RouteStation : public BaseConsoleGenesysApplication {
public:
    Smart_RouteStation();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_ROUTESTATION_H */

```

8.111 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelay↩ Release.cpp File Reference

```
#include "Smart_SeizeDelayRelease.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_SeizeDelayRelease.cpp:

8.112 Smart_SeizeDelayRelease.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SeizeDelayRelease.cpp
 * Author: rlcancian
 *
 * Created on 24 de Setembro de 2019, 16:43
 */

#include "Smart_SeizeDelayRelease.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

```

```
Smart_SeizeDelayRelease::Smart_SeizeDelayRelease() {
}

int Smart_SeizeDelayRelease::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    // Handle traces and simulation events to output them
    // insert plugins
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    Model* model = genesys->getModels()->newModel();
    if (false) {
        // build the simulation model
        // set model infos
        ModelSimulation* sim = model->getSimulation();
        sim->setReplicationLength(120);
        sim->setReplicationLengthTimeUnit(Util::TimeUnit::second);
        sim->setNumberOfReplications(2);
        //
        EntityType* customer = new EntityType(model, "Customer");
        //
        Create* create1 = new Create(model);
        create1->setEntityType(customer);
        create1->setTimeBetweenCreationsExpression("expo(20)");
        create1->setTimeUnit(Util::TimeUnit::second);
        create1->setEntitiesPerCreation(1);
        create1->setFirstCreation(0.0);
        //
        Resource* machine1 = new Resource(model, "Machine_1");
        machine1->setCapacity(1);
        //
        Queue* queueSeize1 = new Queue(model, "Seize_1.Queue");
        queueSeize1->setOrderRule(Queue::OrderRule::FIFO);
        //
        Seize* seize1 = new Seize(model);
        seize1->getSeizeRequests()->insert(new SeizableItem(machine1, "1"));
        seize1->setQueue(queueSeize1);
        //
        Delay* delay1 = new Delay(model);
        delay1->setDelayExpression("unif(15,30)");
        delay1->setDelayTimeUnit(Util::TimeUnit::second);
        //
        Release* release1 = new Release(model);
        release1->getReleaseRequests()->insert(new SeizableItem(machine1, "1"));
        //
        Dispose* dispose1 = new Dispose(model);
        // connect model components to create a "workflow"
        create1->getConnections()->insert(seize1);
        seize1->getConnections()->insert(delay1);
        delay1->getConnections()->insert(release1);
        release1->getConnections()->insert(dispose1);
        // save the model into a text file
        model->save("./models/Smart_SeizeDelayRelease.txt");
        // execute the simulation
        sim->start();
    } else {
        model->load("./models/Smart_SeizeDelayRelease.txt");
        model->show();
        model->getSimulation()->start();
    }

    return 0;
};
```

8.113 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelay↩ Release.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_SeizeDelayRelease.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_SeizeDelayRelease

8.114 Smart_SeizeDelayRelease.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SeizeDelayRelease.h
 * Author: rlcancian
 *
 * Created on 24 de Setembro de 2019, 16:43
 */

#ifndef SMART_SEIZEDELAYRELEASE_H
```

```
#define SMART_SEIZEDELAYRELEASE_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_SeizeDelayRelease : public BaseConsoleGenesysApplication {
public:
    Smart_SeizeDelayRelease();
public:
    virtual int main(int argc, char** argv);
};
#endif /* SMART_SEIZEDELAYRELEASE_H */

```

8.115 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelay↩ ReleaseMany.cpp File Reference

```
#include "Smart_SeizeDelayReleaseMany.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_SeizeDelayReleaseMany.cpp:

8.116 Smart_SeizeDelayReleaseMany.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SeizeDelayReleaseMany.cpp
 * Author: rlcancian
 *
 * Created on 9 de abril de 2021, 18:09
 */

#include "Smart_SeizeDelayReleaseMany.h"

// you have to included need libs

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"

// Model elements
#include "../../kernel/simulator/EntityType.h"

Smart_SeizeDelayReleaseMany::Smart_SeizeDelayReleaseMany() {
```

```
}

int Smart_SeizeDelayReleaseMany::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    Model* m = genesys->getModels()->newModel();
    //m->load("./models/Smart_SeizeDelayReleaseMany.txt");
    //genesys->getModels()->current()->getSimulation()->start();
    //return 0;

    EntityType* customer = new EntityType(m, "Customer");
    Create* create1 = new Create(m);
    create1->setEntityType(customer);
    create1->setTimeBetweenCreationsExpression("1");
    Resource* machine1 = new Resource(m);
    Resource* machine2 = new Resource(m);
    Resource* machine3 = new Resource(m);
    Resource* machine4 = new Resource(m);
    Resource* machine5 = new Resource(m);
    Queue* queueSeize1 = new Queue(m);
    queueSeize1->setOrderRule(Queue::OrderRule::FIFO);
    Seize* seize1 = new Seize(m);
    seize1->getSeizeRequests()->insert(new SeizableItem(machine1));
    seize1->getSeizeRequests()->insert(new SeizableItem(machine2));
    seize1->getSeizeRequests()->insert(new SeizableItem(machine3));
    seize1->getSeizeRequests()->insert(new SeizableItem(machine4));
    seize1->getSeizeRequests()->insert(new SeizableItem(machine5));
    seize1->setQueue(queueSeize1);
    Delay* delay1 = new Delay(m);
    delay1->setDelayExpression("unif(0.8,1.2)");
    Release* release1 = new Release(m);
    release1->getReleaseRequests()->insert(new SeizableItem(machine1));
    release1->getReleaseRequests()->insert(new SeizableItem(machine2));
    release1->getReleaseRequests()->insert(new SeizableItem(machine3));
    release1->getReleaseRequests()->insert(new SeizableItem(machine4));
    release1->getReleaseRequests()->insert(new SeizableItem(machine5));
    Dispose* dispose1 = new Dispose(m);
    // connect model components to create a "workflow"
    create1->getConnections()->insert(seize1);
    seize1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(release1);
    release1->getConnections()->insert(dispose1);
    // save the model into a text file
    ModelSimulation* sim = m->getSimulation();
    sim->setReplicationLength(10);
    sim->setNumberOfReplications(3);
    m->save("./models/Smart_SeizeDelayReleaseMany.txt");
    //sim->start();

    do {
        sim->step();
        std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
    } while (sim->isPaused());
    return 0;
};
```

8.117 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_SeizeDelayReleaseMany.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_SeizeDelayReleaseMany.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_SeizeDelayReleaseMany

8.118 Smart_SeizeDelayReleaseMany.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SeizeDelayReleaseMany.h
 * Author: rlcancian
 *
 * Created on 9 de abril de 2021, 18:09
 */

#ifndef MODEL_SEIZEDELAYRELEASEMANY_H
```

```
#define MODEL_SEIZEDELAYRELEASEMANY_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_SeizeDelayReleaseMany : public BaseConsoleGenesysApplication {
public:
    Smart_SeizeDelayReleaseMany();
public:
    virtual int main(int argc, char** argv);
};

#endif /* MODEL_SEIZEDELAYRELEASEMANY_H */

```

8.119 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Sequence.cpp File Reference

```
#include "Smart_Sequence.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Route.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Leave.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/elements/Sequence.h"
#include "../../plugins/elements/Station.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Smart_Sequence.cpp:

8.120 Smart_Sequence.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Sequence.cpp
 * Author: rlcancian
 *
 * Created on 12 de março de 2021, 15:39
 */

#include "Smart_Sequence.h"

#include "../../kernel/simulator/Simulator.h"

#include "../../plugins/components/Create.h"
#include "../../plugins/components/Route.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Leave.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Assign.h"

#include "../../plugins/elements/Sequence.h"
```

```
#include "../../plugins/elements/Station.h"
#include "../../kernel/simulator/EntityType.h"

Smart_Sequence::Smart_Sequence() {
}

int Smart_Sequence::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L6_arrival);
    //genesys->getModels()->loadModel("./models/Smart_StatationRouteSequence.txt");
    //genesys->getModels()->current()->getSimulation()->start();
    //return;
    Model* m = genesys->getModels()->newModel();
    m->getSimulation()->setReplicationLength(60);
    Create* c1 = new Create(m);
    c1->setEntityType(new EntityType(m));
    c1->setTimeBetweenCreationsExpression("10");
    Station* s1 = new Station(m);
    Route* r0 = new Route(m);
    r0->setRouteDestinationType(Route::DestinationType::BySequence);
//(Route::DestinationType::Station);
    //r0->setStation(s1);
    r0->setRouteTimeExpression("0.1");
    Enter* e1 = new Enter(m);
    e1->setStation(s1);
    Delay* d1 = new Delay(m);
    Station* s2 = new Station(m);
    Leave* l1 = new Leave(m);
    l1->setStation(s1);
    Route* r1 = new Route(m);
    r1->setRouteDestinationType(Route::DestinationType::BySequence);
//(Route::DestinationType::Station);
    //r1->setStation(s2);
    r1->setRouteTimeExpression("0.2");
    Enter* e2 = new Enter(m);
    e2->setStation(s2);
    Delay* d2 = new Delay(m);
    d2->setDelayExpression("2");
    Leave* l2 = new Leave(m);
    l2->setStation(s2);
    Station* s3 = new Station(m);
    Route* r2 = new Route(m);
    r2->setRouteDestinationType(Route::DestinationType::BySequence);
//(Route::DestinationType::Station);
    //r2->setStation(s3);
    r2->setRouteTimeExpression("0.3");
    Enter* e3 = new Enter(m);
    e3->setStation(s3);
    Dispose* dp1 = new Dispose(m);

    Sequence* seq = new Sequence(m);
    seq->getSteps()->insert(new SequenceStep(s1));
    seq->getSteps()->insert(new SequenceStep(s2));
    seq->getSteps()->insert(new SequenceStep(s1));
    seq->getSteps()->insert(new SequenceStep(s1));
    seq->getSteps()->insert(new SequenceStep(s3));

    Assign* a1 = new Assign(m);
    a1->getAssignments()->insert(new Assign::Assignment("Entity.Sequence",
std::to_string(seq->getId())));

    c1->getConnections()->insert(a1);
    a1->getConnections()->insert(r0);
    e1->getConnections()->insert(d1);
    d1->getConnections()->insert(l1);
    l1->getConnections()->insert(r1);
    e2->getConnections()->insert(d2);
    d2->getConnections()->insert(l2);
    l2->getConnections()->insert(r2);
    e3->getConnections()->insert(dp1);

    ModelSimulation* sim = m->getSimulation();
    sim->getBreakpointsOnComponent()->insert(a1);
    sim->getBreakpointsOnComponent()->insert(l2);
    sim->getBreakpointsOnTime()->insert(40.0);
    sim->getBreakpointsOnTime()->insert(20.0);
    sim->setShowReportsAfterReplication(false);
    sim->setShowReportsAfterSimulation(false);

    m->save("./models/Smart_Sequence.txt");

    do {
        sim->start();
    } while (sim->isPaused());
    return 0;
```

```
00110 }
```

8.121 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/smarts/Smart_Sequence.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for Smart_Sequence.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Smart_Sequence

8.122 Smart_Sequence.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_Sequence.h
 * Author: rlcancian
 *
 * Created on 12 de março de 2021, 15:39
 */

#ifndef SMART_SEQUENCE_H
#define SMART_SEQUENCE_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class Smart_Sequence : public BaseConsoleGenesysApplication {
public:
    Smart_Sequence();
public:
    virtual int main(int argc, char** argv);
};

#endif /* SMART_SEQUENCE_H */

```

8.123 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/teaching/AnElectronic↩ AssemblyAndTestSystem.cpp File Reference

```
#include "AnElectronicAssemblyAndTestSystem.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "plugins/components/Decide.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/components/SeizableItem.h"
#include "../../plugins/components/QueueableItem.h"
```

Include dependency graph for AnElectronicAssemblyAndTestSystem.cpp:

8.124 AnElectronicAssemblyAndTestSystem.cpp

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.cc to edit this template
 */

/*
 * File:   Smart_AnElectronicAssemblyAndTestSystem.cpp
 * Author: rlcancian
```

```
 *
 * Created on 6 de dezembro de 2021, 11:09
 */


#include "AnElectronicAssemblyAndTestSystem.h"
// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"
// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "plugins/components/Decide.h"
// Model elements
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/components/SeizableItem.h"
#include "../../plugins/components/QueueableItem.h"

AnElectronicAssemblyAndTestSystem::AnElectronicAssemblyAndTestSystem() {
}

int AnElectronicAssemblyAndTestSystem::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    Model* model = genesys->getModels()->newModel();
    ModelInfo* infos = model->getInfos();
    infos->setName("An Electronic Assembly And Test System");
    infos->setDescription("Example 4.1 from chapter 4 \"Modeling Basic Operations and Inputs\" of the Book \"...\"");
    model->getTracer()->setTraceLevel(Util::TraceLevel::L9_mostDetailed);
    Create* create1 = new Create(model, "Part A Arrive");
    Create* create2 = new Create(model, "Part B Arrive");
    Assign* assign1 = new Assign(model, "Assign Part A Sealer and Arrivel Time");
    Assign* assign2 = new Assign(model, "Assign Part B Sealer and Arrivel Time");
    Seize* seize1 = new Seize(model, "Prep A Process Seize");
    Delay* delay1 = new Delay(model, "Prep A Process Delay");
    Release* release1 = new Release(model, "Prep A Process Relase");
    Seize* seize2 = new Seize(model, "Prep B Process Seize");
    Delay* delay2 = new Delay(model, "Prep B Process Delay");
    Release* release2 = new Release(model, "Prep B Process Releese");
    Seize* seize3 = new Seize(model, "Sealer Seize");
    Delay* delay3 = new Delay(model, "Sealer Delay");
    Release* release3 = new Release(model, "Sealer Release");
    Seize* seize4 = new Seize(model, "Rework Seize");
    Delay* delay4 = new Delay(model, "Rework Delay");
    Release* release4 = new Release(model, "Rework Release");
    Decide* decide1 = new Decide(model, "Failed Sealer Inspector");
    Decide* decide2 = new Decide(model, "Failed Rework Inspection");
    Record* record1 = new Record(model, "Record Screpped Parts");
    Record* record2 = new Record(model, "Record Salvaged Parts");
    Record* record3 = new Record(model, "Recorded Shipped Parts");
    Dispose* dispose1 = new Dispose(model, "Scrapped");
    Dispose* dispose2 = new Dispose(model, "Salvaged");
    Dispose* dispose3 = new Dispose(model, "Shipped");
    // connecting the components
    create1->getConnections()->insert(assign1);
    assign1->getConnections()->insert(seize1);
    seize1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(release1);
    release1->getConnections()->insert(seize3);
    seize3->getConnections()->insert(delay3);
    delay3->getConnections()->insert(release3);
    release3->getConnections()->insert(decide1);
    create2->getConnections()->insert(assign2);
    assign2->getConnections()->insert(seize2);
    seize2->getConnections()->insert(delay2);
    delay2->getConnections()->insert(release2);
    release2->getConnections()->insert(seize3);
    decide1->getConnections()->insert(seize4);
    seize4->getConnections()->insert(delay4);
    delay4->getConnections()->insert(release4);
    release4->getConnections()->insert(decide2);
    decide2->getConnections()->insert(record1);
    record1->getConnections()->insert(dispose1);
    decide2->getConnections()->insert(record2);
    record2->getConnections()->insert(dispose2);
    decide1->getConnections()->insert(record3);
    record3->getConnections()->insert(dispose3);
    // creating all ModelElements and filling all components attributes
    EntityType* partA = new EntityType(model, "Part_A");
    create1->setEntityType(partA);
```

```
    create1->setTimeBetweenCreationsExpression("expo(5)");
    create1->setTimeUnit(Util::TimeUnit::minute);
    EntityType* partB = new EntityType(model, "Part_B");
    create2->setEntityType(partB);
    create2->setTimeBetweenCreationsExpression("expo(30)");
    create2->setTimeUnit(Util::TimeUnit::minute);
    new Attribute(model, "Sealer_Time"); // NOT SAVED IN FILE!!
    new Attribute(model, "Arrive_Time");
    assign1->getAssignments()->insert(new Assign::Assignment("Sealer_Time", "tria(1,3,4)"));
    assign1->getAssignments()->insert(new Assign::Assignment("Arrive_Time", "tnow"));
    assign2->getAssignments()->insert(new Assign::Assignment("Sealer_Time", "weib(2.5, 5.3)"));
    assign2->getAssignments()->insert(new Assign::Assignment("Arrive_Time", "tnow"));
    Resource* prepA = new Resource(model, "PrepA");
    Queue* queuePrepA = new Queue(model, "QueuePrepA");
    seize1->getSeizeRequests()->insert(new SeizableItem(prepA, "1"));
    seize1->setQueueableItem(new QueueableItem(queuePrepA));
    delay1->setDelayExpression("tria(1,4,8)");
    delay1->setDelayTimeUnit(Util::TimeUnit::minute);
    release1->getReleaseRequests()->insert(new SeizableItem(prepA, "1"));
    Resource* prepB = new Resource(model, "PrepB");
    Queue* queuePrepB = new Queue(model, "QueuePrepB");
    seize2->getSeizeRequests()->insert(new SeizableItem(prepB, "1"));
    seize2->setQueueableItem(new QueueableItem(queuePrepB));
    delay2->setDelayExpression("tria(3,5,10)");
    delay2->setDelayTimeUnit(Util::TimeUnit::minute);
    release2->getReleaseRequests()->insert(new SeizableItem(prepB, "1"));
    Resource* sealer = new Resource(model, "Sealer");
    Queue* queueSealer = new Queue(model, "QueueSealer");
    seize3->getSeizeRequests()->insert(new SeizableItem(sealer, "1"));
    seize3->setQueueableItem(new QueueableItem(queueSealer));
    delay3->setDelayExpression("Sealer_Time");
    delay3->setDelayTimeUnit(Util::TimeUnit::minute);
    release3->getReleaseRequests()->insert(new SeizableItem(sealer, "1"));
    decide1->getConditions()->insert("unif(0,1)<0.09");
    Resource* rework = new Resource(model, "Rework");
    Queue* queueRework = new Queue(model, "QueueRework");
    seize4->getSeizeRequests()->insert(new SeizableItem(rework, "1"));
    seize4->setQueueableItem(new QueueableItem(queueRework));
    delay4->setDelayExpression("45");
    delay4->setDelayTimeUnit(Util::TimeUnit::minute);
    release4->getReleaseRequests()->insert(new SeizableItem(rework, "1"));
    decide2->getConditions()->insert("unif(0,1)<0.20");
    record1->setExpression("tnow - Arrive_Time");
    record1->setExpressionName("Scrapped Parts Total Time");
    record2->setExpression("tnow - Arrive_Time");
    record2->setExpressionName("Salvaged Parts Total Time");
    record3->setExpression("tnow - Arrive_Time");
    record3->setExpressionName("Shipped Parts Total Time");
    //setting model information
    model->getInfos()->setName("An Electronic Assembly and Test System");
    model->getInfos()->setDescription("B14 Chap 04 Modeling Basic Operations ans Inputs. Example
section 4.1");
    //setting simulation parameters
    ModelSimulation* sim = model->getSimulation();
    sim->setReplicationLength(8);
    sim->setReplicationLengthTimeUnit(Util::TimeUnit::hour);
    sim->setWarmUpPeriod(1);
    sim->setWarmUpPeriodTimeUnit(Util::TimeUnit::hour);
    sim->setReplicationReportBaseTimeUnit(Util::TimeUnit::minute);
    //save the model
    model->save("./models/Smart_AnElectronicAssemblyAndTestSystem.txt");
    //simulating the model
    sim->start();

}
```

8.125 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/teaching/AnElectronic↩ AssemblyAndTestSystem.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for AnElectronicAssemblyAndTestSystem.h:

This graph shows which files directly or indirectly include this file:

Classes

- class AnElectronicAssemblyAndTestSystem

8.126 AnElectronicAssemblyAndTestSystem.h

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.h to edit this template
 */

/*
 * File:   Smart_AnElectronicAssemblyAndTestSystem.h
 * Author: rlcancian
 *
 * Created on 6 de dezembro de 2021, 11:09
 */

#ifndef ANELECTRONICASSEMBLYANDTESTSYSTEM_H
#define ANELECTRONICASSEMBLYANDTESTSYSTEM_H
```

```
#include "../../applications/BaseConsoleGenesysApplication.h"

class AnElectronicAssemblyAndTestSystem : public BaseConsoleGenesysApplication {
public:
    AnElectronicAssemblyAndTestSystem();
public:
    virtual int main(int argc, char** argv);

};

#endif /* ANELECTRONICASSEMBLYANDTESTSYSTEM_H */
```

8.127 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESySS/2019_2022_GenESySS/examples/teaching/FullSimulationOf↩ ComplexModel.cpp File Reference

```
#include "FullSimulationOfComplexModel.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Decide.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/elements/Variable.h"
```

Include dependency graph for FullSimulationOfComplexModel.cpp:

8.128 FullSimulationOfComplexModel.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

#include "FullSimulationOfComplexModel.h"

// GEnSyS Simulator
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/simulator/ElementManager.h"

// Model Components
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Decide.h"

// Model elements
```

```
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/elements/Variable.h"

FullSimulationOfComplexModel::FullSimulationOfComplexModel() {

}

int FullSimulationOfComplexModel::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    // insert "fake plugins" since plugins based on dynamic loaded library are not implemented yet
    this->insertFakePluginsByHand(genesys);
    // creates an empty model
    Model* model = genesys->getModels()->newModel();
    // Handle traces and simulation events to output them
    TraceManager* tm = model->getTracer();
    this->setDefaultTraceHandlers(tm);
    // get easy access to classes used to insert components and elements into a model
    ComponentManager* components = model->getComponents();
    //
    // build the simulation model
    //
    ModelInfo* infos = model->getInfos();
    infos->setAnalystName("Your name");
    infos->setProjectTitle("The title of the project");
    infos->setDescription("This simulation model tests the components and elements that have been implemented so far.");

    ModelSimulation* sim = model->getSimulation();
    sim->setReplicationLength(1e4);
    sim->setReplicationLengthTimeUnit(Util::TimeUnit::minute);
    sim->setNumberOfReplications(3000);
    tm->setTraceLevel(Util::TraceLevel::L2_results);

    EntityType* entityType1 = new EntityType(model, "Representative_EntityType");

    Create* create1 = new Create(model);
    create1->setEntityType(entityType1);
    create1->setTimeBetweenCreationsExpression("EXPO(5)");
    create1->setTimeUnit(Util::TimeUnit::minute);
    create1->setEntitiesPerCreation(1);
    components->insert(create1);

    new Attribute(model, "Attribute_1");
    new Variable(model, "Variable_1");

    Assign* assign1 = new Assign(model);
    Assign::Assignment* attrib2Assignment = new Assign::Assignment("Variable_1", "Variable_1 + 1");
    assign1->getAssignments()->insert(attrib2Assignment);
    Assign::Assignment* attrib1Assignment = new Assign::Assignment("Attribute_1", "Variable_1");
    assign1->getAssignments()->insert(attrib1Assignment);

    Decide* decide1 = new Decide(model);
    decide1->getConditions()->insert("UNIF(0,1) > 0.5");

    Resource* maquina1 = new Resource(model, "Máquina 1");
    maquina1->setCapacity(1);

    Queue* filaSeize1 = new Queue(model);
    filaSeize1->setOrderRule(Queue::OrderRule::FIFO);

    Seize* seize1 = new Seize(model);
    seize1->getSeizeRequests()->insert(new SeizableItem(maquina1));
    seize1->setQueue(filaSeize1);

    Delay* delay1 = new Delay(model);
    delay1->setDelayExpression("NORM(5, 3)");
    delay1->setDelayTimeUnit(Util::TimeUnit::minute);

    Release* release1 = new Release(model);
    release1->getReleaseRequests()->insert(new SeizableItem(maquina1));

    Record* record1 = new Record(model);
    record1->setExpressionName("Tempo total no sistema");
    record1->setExpression("TNOW - Entity.ArrivalTime");
    record1->setFilename("./temp/TotalTimeInSystem.txt");

    Dispose* dispose1 = new Dispose(model);

    // connect model components to create a "workflow" -- should always start from a SourceModelComponent and end at a SinkModelComponent (it will be checked)
    create1->getConnections()->insert(assign1);
    assign1->getConnections()->insert(decide1);
    decide1->getConnections()->insert(seize1);
    decide1->getConnections()->insert(dispose1);
    seize1->getConnections()->insert(delay1);
    delay1->getConnections()->insert(release1);
```

```
    release1->getConnections()->insert(record1);
    record1->getConnections()->insert(dispose1);
    // then save the model into a text file
    model->save("./models/AssignWrite3Seizes.txt");
    // execute the simulation
    model->getSimulation()->start();

    return 0;
};
```

8.129 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/teaching/FullSimulationOf↩ ComplexModel.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for FullSimulationOfComplexModel.h:

This graph shows which files directly or indirectly include this file:

Classes

- class FullSimulationOfComplexModel

8.130 FullSimulationOfComplexModel.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   buildSimpleModel1.h
 * Author: rafael.luiz.cancian
 *
 * Created on 2 de Outubro de 2018, 19:18
 */

#ifndef COMPLEXMODEL01_H
#define COMPLEXMODEL01_H

#include "../../applications/BaseConsoleGenesysApplication.h"

class FullSimulationOfComplexModel : public BaseConsoleGenesysApplication {
public:
    FullSimulationOfComplexModel();
public:
    virtual int main(int argc, char** argv);
};

#endif /* COMPLEXMODEL01_H */

```

8.131 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/teaching/OperatingSystem02.cpp File Reference

```
#include "OperatingSystem02.h"
#include "../../kernel/simulator/Simulator.h"
```

```
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../plugins/elements/Queue.h"
#include "../../plugins/elements/Resource.h"
#include "../../plugins/elements/Station.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Route.h"
```

Include dependency graph for OperatingSystem02.cpp:

8.132 OperatingSystem02.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Smart_SistemasOperacionais02.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 12 de abril de 2021, 21:54
00012  */
00013
00014 #include "OperatingSystem02.h"
00015
00016 #include "../../kernel/simulator/Simulator.h"
00017 #include "../../kernel/simulator/Model.h"
00018 #include "../../kernel/simulator/Attribute.h"
00019 #include "../../kernel/simulator/EntityType.h"
00020
00021 #include "../../plugins/elements/Queue.h"
00022 #include "../../plugins/elements/Resource.h"
00023 #include "../../plugins/elements/Station.h"
00024
00025 #include "../../plugins/components/Create.h"
00026 #include "../../plugins/components/Seize.h"
00027 #include "../../plugins/components/Delay.h"
00028 #include "../../plugins/components/Release.h"
00029 #include "../../plugins/components/Dispose.h"
00030 #include "../../plugins/components/Assign.h"
00031 #include "../../plugins/components/Decide.h"
00032 #include "../../plugins/components/Enter.h"
00033 #include "../../plugins/components/Route.h"
00034
00035 OperatingSystem02::OperatingSystem02() {
00036 }
00037
00038 int OperatingSystem02::main(int argc, char** argv) {
00039     Simulator* genesys = new Simulator();
00040     this->insertFakePluginsByHand(genesys);
00041     this->setDefaultTraceHandlers(genesys->getTracer());
00042     Model* model = genesys->getModels()->newModel();
00043     //
00044     // CREATE Processo é criado no computador
00045     EntityType* et = new EntityType(model, "processo");
00046     Create* createProc = new Create(model);
00047     createProc->setDescription("Processo é criado no computador");
00048     createProc->setEntityType(et);
00049     createProc->setTimeBetweenCreationsExpression("expo(10)");
```

```
    createProc->setTimeUnit(Util::TimeUnit::milisecond);
    //
    // ASSIGN Define tempo de execução e memória ocupada
    Assign* assignProc = new Assign(model);
    assignProc->setDescription("Define tempo de execução e memória ocupada");
    assignProc->getAssignments()->insert(new Assign::Assignment("memoriaOcupada", "TRUNC(UNIF(2,32))"));
    assignProc->getAssignments()->insert(new Assign::Assignment("tempoExecucao", "NORM(5,1) * memoriaOcupada/10"));
    // atributos (não precisa guardar objetos se não for usar)
    new Attribute(model, "memoriaOcupada");
    new Attribute(model, "tempoExecucao");

    createProc->getConnections()->insert(assignProc);
    //
    // SEIZE memoria
    Resource* memoria = new Resource(model, "memoria");
    memoria->setCapacity(64);
    Queue* queueMem = new Queue(model, "Fila_Alocacao_Memoria");
    Seize* seizeMem = new Seize(model);
    seizeMem->setDescription("Processo aloca memória");
    seizeMem->setQueue(queueMem);
    seizeMem->getSeizeRequests()->insert(new SeizableItem(memoria, "memoriaOcupada"));

    assignProc->getConnections()->insert(seizeMem);
    //
    // ROUTE Processo é enviado para execução na CPU
    Station* stationExecucao = new Station(model, "Estacao_de_Execucao");
    Route* routeExecucao1 = new Route(model);
    routeExecucao1->setDescription("Processo é enviado para execução na CPU");
    routeExecucao1->setStation(stationExecucao);

    seizeMem->getConnections()->insert(routeExecucao1);
    //
    // ENTER Processo chega para ser executado
    Enter* enterStationExecucao = new Enter(model);
    enterStationExecucao->setDescription("Processo chega para ser executado");
    enterStationExecucao->setStation(stationExecucao);
    //
    // SEIZE Processo aloca CPU
    Resource* cpu = new Resource(model, "CPU");
    Queue* queueCPU = new Queue(model, "Fila_CPU");
    Seize* seizeCPU = new Seize(model);
    seizeCPU->setDescription("Processo aloca CPU");
    seizeCPU->setQueue(queueCPU);
    seizeCPU->getSeizeRequests()->insert(new SeizableItem(cpu, "1"));

    enterStationExecucao->getConnections()->insert(seizeCPU);
    //
    // DECIDE Define tempo de execução da fatia de tempo atual
    Decide* decideFatiaTempo = new Decide(model);
    decideFatiaTempo->setDescription("Define tempo de execução da fatia de tempo atual");
    decideFatiaTempo->getConditions()->insert("tempoExecucao >= 1");

    seizeCPU->getConnections()->insert(decideFatiaTempo);
    //
    // ASSIGN Define execucao por um quantum de tempo
    Assign* assignExecFatia = new Assign(model);
    assignExecFatia->setDescription("Define execucao por um quantum de tempo");
    assignExecFatia->getAssignments()->insert(new Assign::Assignment("fatiaTempo", "1"));
    assignExecFatia->getAssignments()->insert(new Assign::Assignment("tempoExecucao", "tempoExecucao-fatiaTempo"));
    new Attribute(model, "fatiaTempo");

    decideFatiaTempo->getConnections()->insert(assignExecFatia);
    //
    // ASSIGN Executa até o final
    Assign* assignExecAteFinal = new Assign(model);
    assignExecAteFinal->setDescription("Executa até o final");
    assignExecAteFinal->getAssignments()->insert(new Assign::Assignment("fatiaTempo", "tempoExecucao"));
    assignExecAteFinal->getAssignments()->insert(new Assign::Assignment("tempoExecucao", "tempoExecucao - fatiaTempo"));

    decideFatiaTempo->getConnections()->insert(assignExecAteFinal);
    //
    // DELAY Processo executa na CPU
    Delay* delayProcExecCPU = new Delay(model);
    delayProcExecCPU->setDescription("Processo executa na CPU");
    delayProcExecCPU->setDelayExpression("fatiaTempo");
    delayProcExecCPU->setDelayTimeUnit(Util::TimeUnit::milisecond);

    assignExecFatia->getConnections()->insert(delayProcExecCPU);
    assignExecAteFinal->getConnections()->insert(delayProcExecCPU);
    //
    // RELEASE Processo libera CPU
    Release* releaseCPU = new Release(model);
```

```
    releaseCPU->setDescription("Processo libera CPU");
    releaseCPU->getReleaseRequests()->insert(new SeizableItem(cpu, "1"));

    delayProcExecCPU->getConnections()->insert(releaseCPU);
    //
    // DECIDE Se processo ainda precisa executar então vai para estação de execução
    Decide* decideAindaExec = new Decide(model);
    decideAindaExec->setDescription("Se processo ainda precisa executar então vai para estação de execução");
    decideAindaExec->getConditions()->insert("tempoExecucao > 0");

    releaseCPU->getConnections()->insert(decideAindaExec);
    //
    // ROUTE Processo é enviado de volta para execução
    Route* routeExecucao2 = new Route(model);
    routeExecucao2->setDescription("Processo é enviado de volta para execução");
    routeExecucao2->setStation(stationExecucao);

    decideAindaExec->getConnections()->insert(routeExecucao2);
    // ROUTE Processo é enviado para liberar memória
    Station* stationLiberaMem = new Station(model, "Estacao_de_liberacao_de_memoria");
    Route* routeLiberaMem = new Route(model);
    routeLiberaMem->setDescription("Processo é enviado para liberar memória");
    routeLiberaMem->setStation(stationLiberaMem);

    decideAindaExec->getConnections()->insert(routeLiberaMem);
    //
    // ENTER Processo chega para liberar memória
    Enter* enterStationLibera = new Enter(model);
    enterStationLibera->setDescription("Processo chega para liberar memória");
    enterStationLibera->setStation(stationLiberaMem);
    //
    // RELEASE  Processo libera memória
    Release* releaseMem = new Release(model);
    releaseMem->setDescription("Processo libera memória");
    releaseMem->getReleaseRequests()->insert(new SeizableItem(memoria, "memoriaOcupada"));

    enterStationLibera->getConnections()->insert(releaseMem);
    //
    // DISPOSE Processo é encerrado
    Dispose* disp1 = new Dispose(model);
    disp1->setDescription("Processo é encerrado");

    releaseMem->getConnections()->insert(disp1);

    //
    // AJUSTA EXPERIMENTO (MODEL SIMULATION)
    ModelSimulation* sim = model->getSimulation();
    sim->setReplicationLength(1e3);
    sim->setReplicationLengthTimeUnit(Util::TimeUnit::milisecond);
    sim->setNumberOfReplications(30);
    sim->setPauseOnReplication(true);
    // AJUSTA NÍVEL DE TRACE
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L2_results);
    // SALVA MODELO
    model->save("./models/SistemaOperacional02.txt");
    // EXECUTA CADA REPLICAÇÃO, RECOMEÇANDO APÓS CADA PAUSA
    do {
        sim->start();
    } while (sim->isPaused());
    return 0;
}
```

8.133 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/examples/teaching/Operating↩ System02.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for OperatingSystem02.h:

This graph shows which files directly or indirectly include this file:

Classes

- class OperatingSystem02

8.134 OperatingSystem02.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SistemasOperacionais02.h
 * Author: rlcancian
 *
 * Created on 12 de abril de 2021, 21:54
 */

#ifndef OPERATINGSYSTEMSS02_H
#define OPERATINGSYSTEMSS02_H

```

```
#include "../../applications/BaseConsoleGenesysApplication.h"

class OperatingSystem02 : public BaseConsoleGenesysApplication {
public:
    OperatingSystem02();
public:
    virtual int main(int argc, char** argv);
};

#endif /* OPERATINGSYSTEMSS02_H */

```

8.135 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/teaching/OperatingSystem03.cpp File Reference

```
#include "OperatingSystem03.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../plugins/elements/Queue.h"
#include "../../plugins/elements/Resource.h"
#include "../../plugins/elements/Station.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Route.h"
#include <iostream>
```

Include dependency graph for OperatingSystem03.cpp:

8.136 OperatingSystem03.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SistemaOperacional03.cpp
 * Author: rlcancian
 *
 * Created on 23 de abril de 2021, 14:03
 */

#include "OperatingSystem03.h"

#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"

#include "../../plugins/elements/Queue.h"
#include "../../plugins/elements/Resource.h"
#include "../../plugins/elements/Station.h"

#include "../../plugins/components/Create.h"
```

```
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Route.h"

#include <iostream>

OperatingSystem03::OperatingSystem03() {
}

int OperatingSystem03::main(int argc, char** argv) {
    Simulator* genesys = new Simulator();
    this->insertFakePluginsByHand(genesys);
    this->setDefaultTraceHandlers(genesys->getTracer());
    Model* model = genesys->getModels()->newModel();
    //
    // CREATE Processo é criado no computador
    EntityType* et = new EntityType(model, "processo");
    Create* createProc = new Create(model);
    createProc->setDescription("Processo é criado no computador");
    createProc->setEntityType(et);
    createProc->setTimeBetweenCreationsExpression("expo(4)");
    createProc->setTimeUnit(Util::TimeUnit::milisecond);
    //
    // ASSIGN Define informacoes do processo
    Assign* assignProc = new Assign(model);
    assignProc->setDescription("Define informacoes do processo");
    assignProc->getAssignments()->insert(new Assign::Assignment("memoriaOcupada", "TRUNC(UNIF(2,32))"));
    assignProc->getAssignments()->insert(new Assign::Assignment("tempoExecucao", "NORM(6,2) * memoriaOcupada/5"));
    assignProc->getAssignments()->insert(new Assign::Assignment("processoSO", "UNIF(0,1) < 0.1"));
    // atributos (não precisa guardar objetos se não for usar)
    new Attribute(model, "memoriaOcupada");
    new Attribute(model, "tempoExecucao");
    new Attribute(model, "processoSO");

    createProc->getConnections()->insert(assignProc);
    //
    // SEIZE memoria
    Resource* memoria = new Resource(model, "memoria");
    memoria->setCapacity(256);
    Queue* queueMem = new Queue(model, "Fila_Alocacao_Memoria");
    Seize* seizeMem = new Seize(model);
    seizeMem->setDescription("Processo aloca memória");
    seizeMem->setQueue(queueMem);
    seizeMem->getSeizeRequests()->insert(new SeizableItem(memoria, "memoriaOcupada"));

    assignProc->getConnections()->insert(seizeMem);
    //
    // DECIDE Define nucleo execucao do processo
    Decide* decideNucleo = new Decide(model);
    decideNucleo->setDescription("Define nucleo execucao do processo");
    decideNucleo->getConditions()->insert("processoSO == 1");
    decideNucleo->getConditions()->insert("NQ(filaNucleo1) <= NQ(filaNucleo2) and NQ(filaNucleo1) <= NQ(filaNucleo3)");
    decideNucleo->getConditions()->insert("NQ(filaNucleo2) <= NQ(filaNucleo1) and NQ(filaNucleo2) <= NQ(filaNucleo3)");

    seizeMem->getConnections()->insert(decideNucleo);
    //
    // ASSIGN Nucleo 0
    Assign* assignDefNucleo0 = new Assign(model);
    assignDefNucleo0->getAssignments()->insert(new Assign::Assignment("nucleoExecucao", "0"));
    new Attribute(model, "nucleoExecucao");
    //
    // ASSIGN Nucleo 1
    Assign* assignDefNucleo1 = new Assign(model);
    assignDefNucleo1->getAssignments()->insert(new Assign::Assignment("nucleoExecucao", "1"));
    //
    // ASSIGN Nucleo 2
    Assign* assignDefNucleo2 = new Assign(model);
    assignDefNucleo2->getAssignments()->insert(new Assign::Assignment("nucleoExecucao", "2"));
    //
    // ASSIGN Nucleo 3
    Assign* assignDefNucleo3 = new Assign(model);
    assignDefNucleo3->getAssignments()->insert(new Assign::Assignment("nucleoExecucao", "3"));

    decideNucleo->getConnections()->insert(assignDefNucleo0);
    decideNucleo->getConnections()->insert(assignDefNucleo1);
    decideNucleo->getConnections()->insert(assignDefNucleo2);
    decideNucleo->getConnections()->insert(assignDefNucleo3);
    //
```

```
    // ROUTE Processo é enviado para execução na CPU
    Station* stationExecucao = new Station(model, "Estacao_de_Execucao");
    Route* routeExecucao1 = new Route(model);
    routeExecucao1->setDescription("Processo é enviado para execução na CPU");
    routeExecucao1->setStation(stationExecucao);

    assignDefNucleo0->getConnections()->insert(routeExecucao1);
    assignDefNucleo1->getConnections()->insert(routeExecucao1);
    assignDefNucleo2->getConnections()->insert(routeExecucao1);
    assignDefNucleo3->getConnections()->insert(routeExecucao1);

    //
    // ENTER Processo chega para ser executado
    Enter* enterStationExecucao = new Enter(model);
    enterStationExecucao->setDescription("Processo chega para ser executado");
    enterStationExecucao->setStation(stationExecucao);
    //
    // SEIZE Processo aloca Núcleo de Execução
    //     SET RESOURCES
    Set* setNucleos = new Set(model, "Set_Nucleos");
    setNucleos->setSetOfType(Util::TypeOf<Resource>());
    setNucleos->getElementSet()->insert(new Resource(model, "nucleo0"));
    setNucleos->getElementSet()->insert(new Resource(model, "nucleo1"));
    setNucleos->getElementSet()->insert(new Resource(model, "nucleo2"));
    setNucleos->getElementSet()->insert(new Resource(model, "nucleo3"));
    //std::cout « setNucleos->getElementSet()->show() « std::endl;
    //     SET QUEUES
    Set* setFilas = new Set(model, "Set_Filas");
    setFilas->setSetOfType(Util::TypeOf<Queue>());
    setFilas->getElementSet()->insert(new Queue(model, "filaNucleo0"));
    setFilas->getElementSet()->insert(new Queue(model, "filaNucleo1"));
    setFilas->getElementSet()->insert(new Queue(model, "filaNucleo2"));
    setFilas->getElementSet()->insert(new Queue(model, "filaNucleo3"));
    std::cout « setFilas->getElementSet()->show() « std::endl;
    // SEIZE
    Seize* seizeNucleo = new Seize(model);
    seizeNucleo->setDescription("Processo aloca Núcleo de Execução");
    seizeNucleo->setQueueableItem(new QueueableItem(setFilas, QueueableItem::QueueableType::SET,
"nucleoExecucao"));
    seizeNucleo->getSeizeRequests()->insert(new SeizableItem(setNucleos, "1",
SeizableItem::SelectionRule::SPECIFICMEMBER, "", "nucleoExecucao"));

    enterStationExecucao->getConnections()->insert(seizeNucleo);
    //
    // DECIDE Define tempo de execução da fatia de tempo atual
    Decide* decideFatiaTempo = new Decide(model);
    decideFatiaTempo->setDescription("Define tempo de execução da fatia de tempo atual");
    decideFatiaTempo->getConditions()->insert("tempoExecucao >= 2");

    seizeNucleo->getConnections()->insert(decideFatiaTempo);
    //
    // ASSIGN Define execucao por um quantum de tempo
    Assign* assignExecFatia = new Assign(model);
    assignExecFatia->setDescription("Define execucao por um quantum de tempo");
    assignExecFatia->getAssignments()->insert(new Assign::Assignment("fatiaTempo", "2"));
    assignExecFatia->getAssignments()->insert(new Assign::Assignment("tempoExecucao",
"tempoExecucao-fatiaTempo"));
    new Attribute(model, "fatiaTempo");

    decideFatiaTempo->getConnections()->insert(assignExecFatia);
    //
    // ASSIGN Executa até o final
    Assign* assignExecAteFinal = new Assign(model);
    assignExecAteFinal->setDescription("Executa até o final");
    assignExecAteFinal->getAssignments()->insert(new Assign::Assignment("fatiaTempo",
"tempoExecucao"));
    assignExecAteFinal->getAssignments()->insert(new Assign::Assignment("tempoExecucao",
"tempoExecucao - fatiaTempo"));

    decideFatiaTempo->getConnections()->insert(assignExecAteFinal);
    //
    // DELAY Processo executa no nucleo de execucao
    Delay* delayProcExecNucleo = new Delay(model);
    delayProcExecNucleo->setDescription("Processo executa no nucleo de execucao");
    delayProcExecNucleo->setDelayExpression("fatiaTempo");
    delayProcExecNucleo->setDelayTimeUnit(Util::TimeUnit::milisecond);

    assignExecFatia->getConnections()->insert(delayProcExecNucleo);
    assignExecAteFinal->getConnections()->insert(delayProcExecNucleo);
    //
    // RELEASE Processo libera nucleo execucao
    Release* releaseNucleo = new Release(model);
    releaseNucleo->setDescription("Processo libera nucleo execucao");
    releaseNucleo->getReleaseRequests()->insert(new SeizableItem(setNucleos, "1",
SeizableItem::SelectionRule::SPECIFICMEMBER, "", "nucleoExecucao"));

    delayProcExecNucleo->getConnections()->insert(releaseNucleo);
```

```
    //
    // DECIDE Se processo ainda precisa executar então vai para estação de execução
    Decide* decideAindaExec = new Decide(model);
    decideAindaExec->setDescription("Se processo ainda precisa executar então vai para estação de execução");
    decideAindaExec->getConditions()->insert("tempoExecucao > 0");

    releaseNucleo->getConnections()->insert(decideAindaExec);
    //
    // ROUTE Processo é enviado de volta para execução
    Route* routeExecucao2 = new Route(model);
    routeExecucao2->setDescription("Processo é enviado de volta para execução");
    routeExecucao2->setStation(stationExecucao);

    decideAindaExec->getConnections()->insert(routeExecucao2);
    // ROUTE Processo é enviado para liberar memória
    Station* stationLiberaMem = new Station(model, "Estacao_de_liberacao_de_memoria");
    Route* routeLiberaMem = new Route(model);
    routeLiberaMem->setDescription("Processo é enviado para liberar memória");
    routeLiberaMem->setStation(stationLiberaMem);

    decideAindaExec->getConnections()->insert(routeLiberaMem);
    //
    // ENTER Processo chega para liberar memória
    Enter* enterStationLibera = new Enter(model);
    enterStationLibera->setDescription("Processo chega para liberar memória");
    enterStationLibera->setStation(stationLiberaMem);
    //
    // RELEASE  Processo libera memória
    Release* releaseMem = new Release(model);
    releaseMem->setDescription("Processo libera memória");
    releaseMem->getReleaseRequests()->insert(new SeizableItem(memoria, "memoriaOcupada"));

    enterStationLibera->getConnections()->insert(releaseMem);
    //
    // DISPOSE Processo é encerrado
    Dispose* disp1 = new Dispose(model);
    disp1->setDescription("Processo é encerrado");

    releaseMem->getConnections()->insert(disp1);

    //
    // AJUSTA EXPERIMENTO (MODEL SIMULATION)
    ModelSimulation* sim = model->getSimulation();
    sim->setReplicationLength(2);
    sim->setReplicationLengthTimeUnit(Util::TimeUnit::second);
    sim->setReplicationReportBaseTimeUnit(Util::TimeUnit::milisecond);
    sim->setNumberOfReplications(200);
    sim->setWarmUpPeriod(0.02);
    sim->setWarmUpPeriodTimeUnit(Util::TimeUnit::second);
    sim->setShowReportsAfterReplication(true);
    sim->setShowReportsAfterSimulation(true);
    //sim->getBreakpointsOnComponent()->insert(routeExecucao1);
    //sim->getBreakpointsOnComponent()->insert(assignDefNucleo0);
    //sim->getBreakpointsOnComponent()->insert(assignDefNucleo1);
    //sim->getBreakpointsOnComponent()->insert(assignDefNucleo2);
    //sim->getBreakpointsOnComponent()->insert(assignDefNucleo3);
    model->save("./models/SistemaOperacional03.txt");
    genesys->getTracer()->setTraceLevel(Util::TraceLevel::L2_results); // :L9_mostDetailed);
    do {
        sim->start(); //step();
        if (sim->isPaused()) {
            std::cout « "Press ENTER to continue...";
            std::cin.ignore(std::numeric_limits <std::streamsize> ::max(), '\n');
        }
    } while (sim->isPaused());
    return 0;
}
```

8.137 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/examples/teaching/OperatingSystem03.h File Reference

```
#include "../../applications/BaseConsoleGenesysApplication.h"
```

Include dependency graph for OperatingSystem03.h:

This graph shows which files directly or indirectly include this file:

Classes

- class OperatingSystem03

8.138 OperatingSystem03.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Smart_SistemaOperacional03.h
 * Author: rlcancian
 *
 * Created on 23 de abril de 2021, 14:03
 */

#ifndef OPERATINGSYSTEMS03_H
#define OPERATINGSYSTEMS03_H

```

```
#include "../../applications/BaseConsoleGenesysApplication.h"

class OperatingSystem03 : public BaseConsoleGenesysApplication {
public:
    OperatingSystem03();
public:
    virtual int main(int argc, char** argv);
private:

};

#endif /* OPERATINGSYSTEMS03_H */

```

8.139 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Attribute.cpp File Reference

`#include "Attribute.h"`

Include dependency graph for Attribute.cpp:

8.140 Attribute.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Attribute.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 25 de Setembro de 2018, 16:37
 */

#include "Attribute.h"

//using namespace GenesysKernel;

Attribute::Attribute(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Attribute>(), name) {
}

std::string Attribute::show() {
    return ModelElement::show();
}

bool Attribute::_loadInstance(std::map<std::string, std::string>* fields) {
    return ModelElement::_loadInstance(fields);
}

PluginInformation* Attribute::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Attribute>(), &Attribute::LoadInstance);
    return info;

```

```
}

ModelElement* Attribute::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Attribute* newElement = new Attribute(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

std::map<std::string, std::string>* Attribute::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<Attribute>());
    return fields;
}

bool Attribute::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}
```

8.141 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Attribute.h File Reference

```
#include <string>
#include <list>
#include "../util/List.h"
#include "ModelElement.h"
#include "ElementManager.h"
#include "Plugin.h"
```

Include dependency graph for Attribute.h:

Classes

- class Attribute

8.142 Attribute.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Attribute.h
 * Author: rafael.luiz.cancian
 *
 * Created on 25 de Setembro de 2018, 16:37
 */

```

```
#ifndef ATTRIBUTE_H
#define ATTRIBUTE_H

#include <string>
#include <list>
#include "../util/List.h"
#include "ModelElement.h"
#include "ElementManager.h"
#include "Plugin.h"

//namespace GenesysKernel {

class Attribute : public ModelElement {
public:
    Attribute(Model* model, std::string name = "");
    virtual ~Attribute() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:
    //List<unsigned int>* _dimensionSizes = new List<unsigned int>();
};
//namespace\\}
#endif /* ATTRIBUTE_H */

```

8.143 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Component↩Manager.cpp File Reference

```
#include "ComponentManager.h"
#include "../util/List.h"
#include "Model.h"
```

Include dependency graph for ComponentManager.cpp:

8.144 ComponentManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
```

```
 * File:   ComponentManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 28 de Maio de 2019, 10:41
 */

#include "ComponentManager.h"
#include "../util/List.h"
#include "Model.h"

//using namespace GenesysKernel;

ComponentManager::ComponentManager(Model* model) {
    _parentModel = model;
    _components = new List<ModelComponent*>();
    _components->setSortFunc([](const ModelComponent* a, const ModelComponent * b) {
        return a->getId() < b->getId();
    });
}

bool ComponentManager::insert(ModelComponent* comp) {
    if (_components->find(comp) == _components->list()->end()) {
        _components->insert(comp);
        _parentModel->getTracer()->trace(Util::TraceLevel::L2_results, "Component \"" + comp->getName() + "\" successfully inserted");
        _hasChanged = true;
        return true;
    }
    _parentModel->getTracer()->trace(Util::TraceLevel::L2_results, "Component \"" + comp->getName() + "\" could not be inserted");
    return false;
}

ModelComponent* ComponentManager::find(std::string name) {
    for (ModelComponent* component : *_components->list()) {
        if (component->getName() == name) {
            return component;
        }
    }
    return nullptr;
}

void ComponentManager::remove(ModelComponent* comp) {
    _components->remove(comp);
    _parentModel->getTracer()->trace(Util::TraceLevel::L2_results, "Component \"" + comp->getName() + "\" successfully removed");
    _hasChanged = true;
}

void ComponentManager::clear() {
    this->_components->clear();
    _hasChanged = true;
}

//ModelComponent* ComponentManager::getComponent(Util::identification id) {
//}

//ModelComponent* ComponentManager::getComponent(std::string name) {
//}

unsigned int ComponentManager::getNumberOfComponents() {
    return _components->size();
}

std::list<ModelComponent*>::iterator ComponentManager::begin() {
    return _components->list()->begin();
}

std::list<ModelComponent*>::iterator ComponentManager::end() {
    return _components->list()->end();
}

ModelComponent* ComponentManager::front() {
    return _components->front();
}

ModelComponent* ComponentManager::next() {
    return _components->next();
}

bool ComponentManager::hasChanged() const {
    if (_hasChanged)
        return _hasChanged;
    for (ModelComponent* component : *_components->list()) {
        if (component->hasChanged()) {
            //_hasChanged = true;
            return true;
```

```
        }
    }
    return false;
}

void ComponentManager::setHasChanged(bool _hasChanged) {
    this->_hasChanged = _hasChanged;
}
```

8.145 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ComponentManager.h File Reference

`#include "ModelComponent.h"`

Include dependency graph for ComponentManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ComponentManager

8.146 ComponentManager.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ComponentManager.h
 * Author: rafael.luiz.cancian
 *
 * Created on 28 de Maio de 2019, 10:41
 */

#ifndef COMPONENTMANAGER_H
#define COMPONENTMANAGER_H

#include "ModelComponent.h"

//namespace GenesysKernel {
//class Model;

```

```
class ComponentManager {
public:
    ComponentManager(Model* model);
    virtual ~ComponentManager() = default;
public:
    bool insert(ModelComponent* comp);
    void remove(ModelComponent* comp);
    ModelComponent* find(std::string name);
    void clear();
    //bool check(ModelComponent* comp, std::string expressionName, std::string* errorMessage);
    //bool check(std::string compName, std::string expressionName, bool mandatory, std::string*
 errorMessage);
public:
    //ModelComponent* getComponent(Util::identification id);
    //ModelComponent* getComponent(std::string name);
    unsigned int getNumberOfComponents();
    std::list<ModelComponent*>::iterator begin();
    std::list<ModelComponent*>::iterator end();
    ModelComponent* front();
    ModelComponent* next();
    bool hasChanged() const;
    void setHasChanged(bool _hasChanged);
    //int getRankOf(std::string name);
public:
private:
    List<ModelComponent*>* _components;
    Model* _parentModel;
    bool _hasChanged = false;
    std::list<ModelComponent*>::iterator _componentIterator;
};
//namespace\\}
#endif /* COMPONENTMANAGER_H */

```

8.147 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Connection↩ Manager.cpp File Reference

`#include "ConnectionManager.h"`

Include dependency graph for ConnectionManager.cpp:

8.148 ConnectionManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ConnectionManager.cpp
 * Author: rlcancian
 *
```

```
 * Created on 1 de Julho de 2019, 18:39
 */

#include "ConnectionManager.h"

//using namespace GenesysKernel;

ConnectionManager::ConnectionManager() {
}

unsigned int ConnectionManager::size() {
    return _nextConnections->size();
}

ModelComponent* ConnectionManager::front() {
    return _nextConnections->front()->first;
}

ModelComponent* ConnectionManager::atRank(unsigned int rank) {
    return _nextConnections->getAtRank(rank)->first;
}

Connection* ConnectionManager::getFrontConnection() {
    return _nextConnections->front();
}

Connection* ConnectionManager::getConnectionAtRank(unsigned int rank) {
    return _nextConnections->getAtRank(rank);
}

void ConnectionManager::insert(ModelComponent* component, unsigned int inputNumber) {
    Connection* connection = new Connection(component, inputNumber);
    insert(connection);
}

void ConnectionManager::insert(Connection* connection) {
    unsigned int rank = _nextConnections->size();
    _nextConnections->setAtRank(rank, connection);
    _currentOutputConnections++;

}

void ConnectionManager::insertAtRank(unsigned int rank, Connection* connection) {
    _nextConnections->setAtRank(rank, connection); // \TODO: it does not work if there is less then rank connections. Model designer responsability?
    _currentOutputConnections++;
}

std::list<Connection*>* ConnectionManager::list() const {
    return _nextConnections->list();
}

//void ConnectionManager::setCurrentOutputConnections(unsigned int _currentOutputConnections) {
//    this->_currentOutputConnections = _currentOutputConnections;
//}

unsigned int ConnectionManager::getCurrentOutputConnections() const {
    return _currentOutputConnections;
}

void ConnectionManager::setMaxOutputConnections(unsigned int _maxOutputConnections) {
    this->_maxOutputConnections = _maxOutputConnections;
}

unsigned int ConnectionManager::getMaxOutputConnections() const {
    return _maxOutputConnections;
}

void ConnectionManager::setMinOutputConnections(unsigned int _minOutputConnections) {
    this->_minOutputConnections = _minOutputConnections;
}

unsigned int ConnectionManager::getMinOutputConnections() const {
    return _minOutputConnections;
}

//void ConnectionManager::setCurrentInputConnections(unsigned int _currentInputConnections) {
//    this->_currentInputConnections = _currentInputConnections;
//}

unsigned int ConnectionManager::getCurrentInputConnections() const {
    return _currentInputConnections;
}

void ConnectionManager::setMaxInputConnections(unsigned int _maxInputConnections) {
    this->_maxInputConnections = _maxInputConnections;
}
```

```

unsigned int ConnectionManager::getMaxInputConnections() const {
    return _maxInputConnections;
}

void ConnectionManager::setMinInputConnections(unsigned int _minInputConnections) {
    this->_minInputConnections = _minInputConnections;
}

unsigned int ConnectionManager::getMinInputConnections() const {
    return _minInputConnections;
}
```

8.149 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ConnectionManager.h File Reference

```
#include <utility>
#include "../util/List.h"
```

Include dependency graph for ConnectionManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ConnectionManager

Typedefs

- typedef std::pair< ModelComponent *, unsigned int > Connection

8.149.1 Typedef Documentation

8.149.1.1 Connection `typedef std::pair<ModelComponent*, unsigned int> Connection`

Definition at line 24 of file ConnectionManager.h.

8.150 ConnectionManager.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ConnectionManager.h
00009  * Author: rlcancian
00010  *
00011  * Created on 1 de Julho de 2019, 18:39
00012  */
00013
00014 #ifndef CONNECTIONMANAGER_H
00015 #define CONNECTIONMANAGER_H
00016
00017 #include <utility>
00018 #include "../util/List.h"
00019
00020 //namespace GenesysKernel {
00021
00022 class ModelComponent;
00023
00024 typedef std::pair<ModelComponent*, unsigned int> Connection;
00025
00029 class ConnectionManager {
00030 public:
00031     ConnectionManager();
00032     virtual ~ConnectionManager() = default;
00033 public:
00034     unsigned int size();
00035     ModelComponent* front();
00036     ModelComponent* atRank(unsigned int rank);
00037     Connection* getFrontConnection();
00038     Connection* getConnectionAtRank(unsigned int rank);
00039     void insert(ModelComponent* component, unsigned int inputNumber = 0);
00040     void insert(Connection* connection);
00041     void insertAtRank(unsigned int rank, Connection* connection);
00042     //void insert(ModelComponent* component, unsigned int inputNumber = 0);
00043     std::list<Connection*>* list() const;
00044     //void setCurrentOutputConnections(unsigned int _currentOutputConnections);
00045     unsigned int getCurrentOutputConnections() const;
00046     void setMaxOutputConnections(unsigned int _maxOutputConnections);
00047     unsigned int getMaxOutputConnections() const;
00048     void setMinOutputConnections(unsigned int _minOutputConnections);
00049     unsigned int getMinOutputConnections() const;
00050     //void setCurrentInputConnections(unsigned int _currentInputConnections);
00051     unsigned int getCurrentInputConnections() const;
00052     void setMaxInputConnections(unsigned int _maxInputConnections);
00053     unsigned int getMaxInputConnections() const;
00054     void setMinInputConnections(unsigned int _minInputConnections);
00055     unsigned int getMinInputConnections() const;
00056 private:
00057     List<Connection*>* _nextConnections = new List<Connection*>();
00058     unsigned int _minInputConnections = 1;
00059     unsigned int _maxInputConnections = 1;
00060     unsigned int _currentInputConnections = 1;
00061     unsigned int _minOutputConnections = 1;
00062     unsigned int _maxOutputConnections = 1;
00063     unsigned int _currentOutputConnections = 1;
00064 };
00065 //namespace\\}
00066 #endif /* CONNECTIONMANAGER_H */
00067
```

8.151 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Counter.cpp File Reference

```
#include "Counter.h"
#include "SimulationResponse.h"
```

#include "Model.h"

Include dependency graph for Counter.cpp:

8.152 Counter.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   CounterDefaultImpl1.cpp
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 29 de Maio de 2019, 11:24
00012  */
00013
00014 #include "Counter.h"
00015 #include "SimulationResponse.h"
00016 #include "Model.h"
00017
00018 //using namespace GenesysKernel;
00019
00020 Counter::Counter(Model* model, std::string name, ModelElement* parent) : ModelElement(model,
       Util::TypeOf<Counter>(), name) {
00021     _parent = parent;
00022     GetterMember getterMember = DefineGetterMember<Counter>(this, &Counter::getCountValue);
00023     std::string parentName = "";
00024     if (_parent != nullptr)
00025         parentName = _parent->getName();
00026     SimulationResponse* resp = new SimulationResponse(Util::TypeOf<Counter>(), /*parentName + ":" +*/
       getName(), getterMember, parentName);
00027     _parentModel->getResponses()->insert(resp);
00028 }
00029
00030 std::string Counter::show() {
00031     return ModelElement::show() +
00032             ", count=" + std::to_string(this->_count);
00033 }
00034
00035 void Counter::clear() {
00036     _count = 0.0;
00037 }
00038
00039 void Counter::incCountValue(double value) {
00040     _count += value;
00041 }
00042
00043 double Counter::getCountValue() const {
00044     return _count;
00045 }
00046
00047 ModelElement* Counter::getParent() const {
00048     return _parent;
00049 }
00050
00051 PluginInformation* Counter::GetPluginInformation() {
00052     PluginInformation* info = new PluginInformation(Util::TypeOf<Counter>(), &Counter::LoadInstance);
00053     info->setGenerateReport(true);
```

```
    return info;
}

ModelElement* Counter::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Counter* newElement = new Counter(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool Counter::_loadInstance(std::map<std::string, std::string>* fields) {
    return ModelElement::_loadInstance(fields);
}

std::map<std::string, std::string>* Counter::_saveInstance() {
    return ModelElement::_saveInstance();
}

bool Counter::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}

void Counter::_initBetweenReplications() {
    _count = 0.0;
}
```

8.153 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Counter.h File Reference

```
#include "ModelElement.h"
#include "ElementManager.h"
#include "Plugin.h"
```

Include dependency graph for Counter.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Counter

8.154 Counter.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CounterDefaultImpl1.h
 * Author: rafael.luiz.cancian
 *
 * Created on 29 de Maio de 2019, 11:24
 */

#ifndef COUNTERDEFAULTIMPL1_H
#define COUNTERDEFAULTIMPL1_H

#include "ModelElement.h"
#include "ElementManager.h"
#include "Plugin.h"
//namespace GenesysKernel {

class Counter : public ModelElement {
public:
    Counter(Model* model, std::string name = "", ModelElement* parent = nullptr);
    virtual ~Counter() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    void clear();
    void incCountValue(/*int*/double value = 1.0);
    double /*unsigned long*/ getCountValue() const;
    ModelElement* getParent() const;
protected: // from ModelElement
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _initBetweenReplications();
private:
    ModelElement* _parent;
    double /*unsigned long*/ _count = 0;
};
//namespace\\}
#endif /* COUNTERDEFAULTIMPL1_H */
```

8.155 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/DefineGetterSetter.h File Reference

```
#include <functional>
#include "../util/Util.h"
```

Include dependency graph for DefineGetterSetter.h:

This graph shows which files directly or indirectly include this file:

Typedefs

- typedef std::function< double() > GetterMember

  Pointer to a getter function that return a double. Used to get values from SimulationControl and Simulationresponse.
- typedef std::function< void(double) > SetterMember

  Pointer to a setter function that sets a double. Used to set values to Simulationresponse.
- typedef std::function< std::string() > GetterMemberString

  Pointer to a getter function that return a string. Used to get values from SimulationControl and Simulationresponse.
- typedef std::function< void(std::string) > SetterMemberString

  Pointer to a setter function that sets a string. Used to set values to Simulationresponse.
- typedef std::function< double() > GetterMemberBool

  Pointer to a getter function that return a bool. Used to get values from SimulationControl and Simulationresponse.
- typedef std::function< void(double) > SetterMemberBool

  Pointer to a setter function that sets a bool. Used to set values to Simulationresponse.

Functions

- template<typename Class >
  GetterMember DefineGetterMember (Class *object, double(Class::*function)())
- template<typename Class >
  SetterMember DefineSetterMember (Class *object, void(Class::*function)(double))
- template<typename Class >
  GetterMember DefineGetterMember (Class *object, unsigned int(Class::*function)() const)
- template<typename Class >
  SetterMember DefineSetterMember (Class *object, void(Class::*function)(unsigned int))
- template<typename Class >
  GetterMember DefineGetterMember (Class *object, unsigned long(Class::*function)() const)
- template<typename Class >
  SetterMember DefineSetterMember (Class *object, void(Class::*function)(unsigned long))
- template<typename Class >
  GetterMember DefineGetterMember (Class *object, bool(Class::*function)() const)
- template<typename Class >
  SetterMember DefineSetterMember (Class *object, void(Class::*function)(bool))
- template<typename Class >
  GetterMember DefineGetterMember (Class *object, std::string(Class::*function)() const)
- template<typename Class >
  SetterMember DefineSetterMember (Class *object, void(Class::*function)(std::string) const)
- template<typename Class >
  GetterMemberString DefineGetterMember (Class *object, std::string(Class::*function)() const)
- template<typename Class >
  SetterMemberString DefineSetterMember (Class *object, void(Class::*function)(std::string) const)
- template<typename Class >
  GetterMember DefineGetterMember (Class *object, Util::TimeUnit(Class::*function)() const)
- template<typename Class >
  SetterMember DefineSetterMember (Class *object, void(Class::*function)(Util::TimeUnit))

8.155.1 Typedef Documentation

8.155.1.1 GetterMember `typedef std::function<double() > GetterMember`

Pointer to a getter function that return a double. Used to get values from SimulationControl and Simulationresponse.

Definition at line 20 of file DefineGetterSetter.h.

8.155.1.2 GetterMemberBool `typedef std::function<double() > GetterMemberBool`

Pointer to a getter function that return a bool. Used to get values from SimulationControl and Simulationresponse.

Definition at line 26 of file DefineGetterSetter.h.

8.155.1.3 GetterMemberString `typedef std::function<std::string() > GetterMemberString`

Pointer to a getter function that return a string. Used to get values from SimulationControl and Simulationresponse.

Definition at line 23 of file DefineGetterSetter.h.

8.155.1.4 SetterMember `typedef std::function<void(double) > SetterMember`

Pointer to a setter function that sets a double. Used to set values to Simulationresponse.

Definition at line 21 of file DefineGetterSetter.h.

8.155.1.5 SetterMemberBool `typedef std::function<void(double) > SetterMemberBool`

Pointer to a setter function that sets a bool. Used to set values to Simulationresponse.

Definition at line 27 of file DefineGetterSetter.h.

8.155.1.6 SetterMemberString `typedef std::function<void(std::string) > SetterMemberString`

Pointer to a setter function that sets a string. Used to set values to Simulationresponse.

Definition at line 24 of file DefineGetterSetter.h.

8.155.2 Function Documentation

8.155.2.1 DefineGetterMember() [1/7] template<typename Class >

```
GetterMember DefineGetterMember (
            Class * object,
            bool(Class::*)() const function )
```

Definition at line 66 of file DefineGetterSetter.h.

8.155.2.2 DefineGetterMember() [2/7] template<typename Class >

```
GetterMember DefineGetterMember (
            Class * object,
            double(Class::*)() function )
```

Definition at line 31 of file DefineGetterSetter.h.

8.155.2.3 DefineGetterMember() [3/7] template<typename Class >

```
GetterMember DefineGetterMember (
            Class * object,
            std::string(Class::*)() const function )
```

Definition at line 76 of file DefineGetterSetter.h.

8.155.2.4 DefineGetterMember() [4/7] template<typename Class >

```
GetterMemberString DefineGetterMember (
            Class * object,
            std::string(Class::*)() const function )
```

Definition at line 84 of file DefineGetterSetter.h.

8.155.2.5 DefineGetterMember() [5/7] template<typename Class >

```
GetterMember DefineGetterMember (
            Class * object,
            unsigned int(Class::*)() const function )
```

Definition at line 45 of file DefineGetterSetter.h.

8.155.2.6 DefineGetterMember() [6/7] template<typename Class >

```
GetterMember DefineGetterMember (
            Class * object,
            unsigned long(Class::*)() const function )
```

Definition at line 56 of file DefineGetterSetter.h.

8.155.2.7 DefineGetterMember() [7/7] template<typename Class >

```
GetterMember DefineGetterMember (
            Class * object,
            Util::TimeUnit(Class::*)() const function )
```

Definition at line 95 of file DefineGetterSetter.h.

8.155.2.8 DefineSetterMember() [1/7] template<typename Class >

```
SetterMember DefineSetterMember (
            Class * object,
            void(Class::*)(bool) function )
```

Definition at line 70 of file DefineGetterSetter.h.

8.155.2.9 DefineSetterMember() [2/7] template<typename Class >

```
SetterMember DefineSetterMember (
            Class * object,
            void(Class::*)(double) function )
```

Definition at line 39 of file DefineGetterSetter.h.

8.155.2.10 DefineSetterMember() [3/7] template<typename Class >

```
SetterMember DefineSetterMember (
            Class * object,
            void(Class::*)(std::string) const function )
```

Definition at line 80 of file DefineGetterSetter.h.

8.155.2.11 DefineSetterMember() [4/7] template<typename Class >

```
SetterMemberString DefineSetterMember (
            Class * object,
            void(Class::*)(std::string) const function )
```

Definition at line 88 of file DefineGetterSetter.h.

8.155.2.12 DefineSetterMember() [5/7] template<typename Class >

```
SetterMember DefineSetterMember (
            Class * object,
            void(Class::*)(unsigned int) function )
```

Definition at line 49 of file DefineGetterSetter.h.

8.155.2.13 DefineSetterMember() [6/7] template<typename Class >

```
SetterMember DefineSetterMember (
            Class * object,
            void(Class::*)(unsigned long) function )
```

Definition at line 60 of file DefineGetterSetter.h.

8.155.2.14 DefineSetterMember() [7/7] template<typename Class >

```
SetterMember DefineSetterMember (
            Class * object,
            void(Class::*)(Util::TimeUnit) function )
```

Definition at line 99 of file DefineGetterSetter.h.

8.156 DefineGetterSetter.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   DefineGetterSetter.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 5 de Agosto de 2018, 13:52
00012  */
00013
00014 #ifndef DEFINEGETTERSETTER_H
00015 #define DEFINEGETTERSETTER_H
00016
00017 #include <functional>
00018 #include "../util/Util.h"
00019
00020 typedef std::function<double() > GetterMember;
00021 typedef std::function<void(double) > SetterMember;
00022
00023 typedef std::function<std::string() > GetterMemberString;
00024 typedef std::function<void(std::string) > SetterMemberString;
00025
00026 typedef std::function<double() > GetterMemberBool;
00027 typedef std::function<void(double) > SetterMemberBool;
00028
00029 // double
00030 template<typename Class>
00031 GetterMember DefineGetterMember(Class * object, double (Class::*function)()) {
00032     return std::bind(function, object);
00033 }
00034 template<typename Class>
00035 GetterMember DefineGetterMember(Class * object, double (Class::*function)() const) {
00036     return std::bind(function, object);
00037 }
00038 template<typename Class>
00039 SetterMember DefineSetterMember(Class * object, void (Class::*function)(double)) {
00040     return std::bind(function, object, std::placeholders::_1);
00041 }
00042
```

```
// unsigned int
template<typename Class>
GetterMember DefineGetterMember(Class * object, unsigned int (Class::*function)() const) {
    return std::bind(function, object);
}
template<typename Class>
SetterMember DefineSetterMember(Class * object, void (Class::*function)(unsigned int)) {
    return std::bind(function, object, std::placeholders::_1);
}


// unsigned long
template<typename Class>
GetterMember DefineGetterMember(Class * object, unsigned long (Class::*function)() const) {
    return std::bind(function, object);
}
template<typename Class>
SetterMember DefineSetterMember(Class * object, void (Class::*function)(unsigned long)) {
    return std::bind(function, object, std::placeholders::_1);
}

// bool
template<typename Class>
GetterMember DefineGetterMember(Class * object, bool (Class::*function)() const) {
    return std::bind(function, object);
}
template<typename Class>
SetterMember DefineSetterMember(Class * object, void (Class::*function)(bool)) {
    return std::bind(function, object, std::placeholders::_1);
}

//std::string
template<typename Class>
GetterMember DefineGetterMember(Class * object, std::string(Class::*function)() const) {
    return std::bind(function, object);
}
template<typename Class>
SetterMember DefineSetterMember(Class * object, void (Class::*function)(std::string) const) {
    return std::bind(function, object, std::placeholders::_1);
}
template<typename Class>
GetterMemberString DefineGetterMember(Class * object, std::string(Class::*function)() const) {
    return std::bind(function, object);
}
template<typename Class>
SetterMemberString DefineSetterMember(Class * object, void (Class::*function)(std::string) const) {
    return std::bind(function, object, std::placeholders::_1);
}


// Util::TimeUnit
template<typename Class>
GetterMember DefineGetterMember(Class * object, /*GenesysKernel::*/Util::TimeUnit(Class::*function)() const) {
    return std::bind(function, object);
}
template<typename Class>
SetterMember DefineSetterMember(Class * object, void (Class::*function)(/*GenesysKernel::*/Util::TimeUnit)) {
    return std::bind(function, object, std::placeholders::_1);
}


#endif /* DEFINEGETTERSETTER_H */

```

8.157 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ElementManager.cpp File Reference

```
#include "ElementManager.h"
#include "Model.h"
```

Include dependency graph for ElementManager.cpp:

8.158 ElementManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ElementManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 7 de Novembro de 2018, 12:48
 */

#include "ElementManager.h"
#include "Model.h"

//using namespace GenesysKernel;

ElementManager::ElementManager(Model* model) {
    _parentModel = model;
    /* \todo: -- Sort methods for elements should be a decorator */
    _elements = new std::map<std::string, List<ModelElement*>*>();
}

bool ElementManager::insert(ModelElement* anElement) {
    std::string elementTypename = anElement->getClassname();
    return insert(elementTypename, anElement);
}

bool ElementManager::insert(std::string elementTypename, ModelElement* anElement) {
    List<ModelElement*>* listElements = getElementList(elementTypename);
    if (listElements->find(anElement) == listElements->list()->end()) { //not found
        listElements->insert(anElement);
        _hasChanged = true;
        this->_parentModel->getTracer()->trace(Util::TraceLevel::L8_detailed, anElement->getClassname() + " \"" + anElement->getName() + "\"" + " successfully inserted.");
        return true;
    } else {
        this->_parentModel->getTracer()->trace(Util::TraceLevel::L8_detailed, anElement->getClassname() + " \"" + anElement->getName() + "\" already exists."); //could not be inserted.");
    }
    return false;
}

void ElementManager::remove(ModelElement* anElement) {
    std::string elementTypename = anElement->getClassname();
    List<ModelElement*>* listElements = getElementList(elementTypename);
    listElements->remove(anElement);
    _hasChanged = true;
    _parentModel->getTracer()->trace(Util::TraceLevel::L2_results, "Element successfully removed.");

}
```

```
00051
00052 void ElementManager::remove(std::string elementTypename, ModelElement* anElement) {
00053     List<ModelElement*>* listElements = getElementList(elementTypename);
00054     _hasChanged = true;
00055     listElements->remove(anElement);
00056 }
00057
00058 bool ElementManager::check(std::string elementTypename, std::string elementName, std::string
      expressionName, bool mandatory, std::string* errorMessage) {
00059     if (elementName == "" && !mandatory) {
00060         return true;
00061     }
00062     bool result = getElement(elementTypename, elementName) != nullptr;
00063     if (!result) {
00064         std::string msg = elementTypename + " \"" + elementName + "\" for '" + expressionName + "' is
      not in the model.";
00065         errorMessage->append(msg);
00066     }
00067     return result;
00068 }
00069
00070 bool ElementManager::check(std::string elementTypename, ModelElement* anElement, std::string
      expressionName, std::string* errorMessage) {
00071     bool result = anElement != nullptr;
00072     if (!result) {
00073         std::string msg = elementTypename + " for '" + expressionName + "' is null.";
00074         errorMessage->append(msg);
00075     } else {
00076         result = check(elementTypename, anElement->getName(), expressionName, true, errorMessage);
00077     }
00078     return result;
00079 }
00080
00081 void ElementManager::clear() {
00082     _hasChanged = true;
00083     this->_elements->clear();
00084 }
00085
00086 unsigned int ElementManager::getNumberOfElements(std::string elementTypename) {
00087     List<ModelElement*>* listElements = getElementList(elementTypename);
00088     return listElements->size();
00089 }
00090
00091 unsigned int ElementManager::getNumberOfElements() {
00092     unsigned int total = 0;
00093     for (std::map<std::string, List<ModelElement*>*>::iterator it = _elements->begin(); it !=
      _elements->end(); it++) {
00094         total += (*it).second->size();
00095     }
00096     return total;
00097 }
00098
00099 void ElementManager::show() {
00100     _parentModel->getTracer()->trace(Util::TraceLevel::L8_detailed, "Model Elements:");
00101     //std::map<std::string, List<ModelElement*>*>* _elements;
00102     std::string key;
00103     List<ModelElement*>* list;
00104     Util::IncIndent();
00105     {
00106         for (std::map<std::string, List<ModelElement*>*>::iterator infraIt = _elements->begin();
      infraIt != _elements->end(); infraIt++) {
00107             key = (*infraIt).first;
00108             list = (*infraIt).second;
00109             _parentModel->getTracer()->trace(Util::TraceLevel::L8_detailed, key + ": (" +
      std::to_string(list->size()) + ")");
00110             Util::IncIndent();
00111             {
00112                 for (std::list<ModelElement*>::iterator it = list->list()->begin(); it !=
      list->list()->end(); it++) {
00113                     _parentModel->getTracer()->trace(Util::TraceLevel::L8_detailed, (*it)->show());
00114                 }
00115             }
00116             Util::DecIndent();
00117         }
00118     }
00119     Util::DecIndent();
00120 }
00121
00122 Model* ElementManager::getParentModel() const {
00123     return _parentModel;
00124 }
00125
00126 bool ElementManager::hasChanged() const {
00127     if (_hasChanged)
00128         return _hasChanged;
00129     else {
00130         for (std::pair<std::string, List<ModelElement*>*> pair : *_elements) {
```

```
            for (ModelElement* element : *pair.second->list()) {
                if (element->hasChanged()) {
                    return true;
                }
            }
        }
    }
    return false;
}

void ElementManager::setHasChanged(bool _hasChanged) {
    this->_hasChanged = _hasChanged;
}

List<ModelElement*>* ElementManager::getElementList(std::string elementTypename) const {
    std::map<std::string, List<ModelElement*>*>::iterator it = this->_elements->find(elementTypename);
    if (it == this->_elements->end()) {
        // list does not exists yet. Create it and set a valid iterator
        List<ModelElement*>* newList = new List<ModelElement*>();
        newList->setSortFunc([](const ModelElement* a, const ModelElement * b) {
            return a->getId() < b->getId();
        });
        _elements->insert(std::pair<std::string, List<ModelElement*>*>(elementTypename, newList));
        it = this->_elements->find(elementTypename);
    }
    List<ModelElement*>* infras = it->second;
    return infras;
}

ModelElement* ElementManager::getElement(std::string elementTypename, Util::identification id) {
    List<ModelElement*>* list = getElementList(elementTypename);
    for (std::list<ModelElement*>::iterator it = list->list()->begin(); it != list->list()->end(); it++) {
        if ((*it)->getId() == id) { // found
            return (*it);
        }
    }
    return nullptr;
}

int ElementManager::getRankOf(std::string elementTypename, std::string name) {
    int rank = 0;
    List<ModelElement*>* list = getElementList(elementTypename);
    for (std::list<ModelElement*>::iterator it = list->list()->begin(); it != list->list()->end(); it++) {
        if ((*it)->getName() == name) { // found
            return rank;
        } else {
            rank++;
        }
    }
    return -1;
}

std::list<std::string>* ElementManager::getElementClassnames() const {
    std::list<std::string>* keys = new std::list<std::string>();
    for (std::map<std::string, List<ModelElement*>*>::iterator it = _elements->begin(); it != _elements->end(); it++) {
        keys->insert(keys->end(), (*it).first);
    }
    return keys;
}

ModelElement* ElementManager::getElement(std::string elementTypename, std::string name) {
    List<ModelElement*>* list = getElementList(elementTypename);
    for (std::list<ModelElement*>::iterator it = list->list()->begin(); it != list->list()->end(); it++) {
        if ((*it)->getName() == name) { // found
            return (*it);
        }
    }
    return nullptr;
}
```

8.159 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ElementManager.h File Reference

```
#include <list>
#include <map>
```

```
#include "../util/List.h"
#include "ModelElement.h"
```

Include dependency graph for ElementManager.h:

Classes

- class ElementManager

8.160 ElementManager.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ElementManager.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 7 de Novembro de 2018, 12:48
00012  */
00013
00014 #ifndef ELEMENTMANAGER_H
00015 #define ELEMENTMANAGER_H
00016
00017 #include <list>
00018 #include <map>
00019 #include "../util/List.h"
00020 #include "ModelElement.h"
00021
00022 //namespace GenesysKernel {
00023 class Model;
00024
00030 class ElementManager {
00031 public:
00032     ElementManager(Model* model);
00033     virtual ~ElementManager() = default;
00034 public:
00035     bool insert(ModelElement* anElement);
00036     void remove(ModelElement* anElement);
00037     bool insert(std::string elementTypename, ModelElement* anElement);
00038     void remove(std::string elementTypename, ModelElement* anElement);
00039     bool check(std::string elementTypename, ModelElement* anElement, std::string expressionName,
       std::string* errorMessage);
00040     bool check(std::string elementTypename, std::string elementName, std::string expressionName, bool
       mandatory, std::string* errorMessage);
00041     void clear();
00042 public:
00043     ModelElement* getElement(std::string elementTypename, Util::identification id);
00044     ModelElement* getElement(std::string elementTypename, std::string name);
00045     unsigned int getNumberOfElements(std::string elementTypename);
00046     unsigned int getNumberOfElements();
00047     int getRankOf(std::string elementTypename, std::string name);
00048     std::list<std::string>* getElementClassnames() const;
00049
00050     //private:
00051 public:
00052     // \todo: Should be removed and replaced by GetElement(elementType, rank)
00053     List<ModelElement*>* getElementList(std::string elementTypename) const;
00054 public:
00055     void show();
00056     Model* getParentModel() const;
```

```
    bool hasChanged() const;
    void setHasChanged(bool _hasChanged);
private:
    std::map<std::string, List<ModelElement*>*>* _elements;
    Model* _parentModel;
    bool _hasChanged = false;
};
//namespace\\}
#endif /* ELEMENTMANAGER_H */

```

8.161 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ElementManager_if.h File Reference

Classes

- class ElementManager_if

8.162 ElementManager_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ElementManager_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Novembro de 2018, 01:06
 */

#ifndef ELEMENTMANAGER_IF_H
#define ELEMENTMANAGER_IF_H

class ElementManager_if {
public:
    ElementManager_if();
    ElementManager_if(const ElementManager_if& orig);
    virtual ~ElementManager_if();
private:

};

#endif /* ELEMENTMANAGER_IF_H */

```

8.163 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Entity.cpp File Reference

```
#include <typeinfo>
#include "Entity.h"
#include "Attribute.h"
#include "Model.h"
```

#include <cassert>

Include dependency graph for Entity.cpp:

8.164 Entity.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Entity.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 16:30
 */

#include <typeinfo>
#include "Entity.h"
#include "Attribute.h"
#include "Model.h"
#include <cassert>

//using namespace GenesysKernel;

Entity::Entity(Model* model, std::string name, bool insertIntoModel) : ModelElement(model, Util::TypeOf<Entity>(), name, insertIntoModel) {
    _entityNumber = Util::GetLastIdOfType(Util::TypeOf<Entity>());
    unsigned int numAttributes = _parentModel->getElements()->getNumberOfElements(Util::TypeOf<Attribute>());
    for (unsigned i = 0; i < numAttributes; i++) {
        std::map<std::string, double>* map = new std::map<std::string, double>();
        _attributeValues->insert(map);
    }
}

void Entity::setEntityTypeName(std::string entityTypeName) throw () {
    EntityType* entitytype = dynamic_cast<EntityType*>(_parentModel->getElements()->getElement(Util::TypeOf<EntityType>(), entityTypeName));
    if (entitytype != nullptr) {
        this->_entityType = entitytype;
    } else {

        throw std::invalid_argument("EntityType does not exist");
    }
}

std::string Entity::getEntityTypeName() const {

    return this->_entityType->getName();
}

void Entity::setEntityType(EntityType* entityType) {

    _entityType = entityType;
}
```

```cpp

EntityType* Entity::getEntityType() const {

    return _entityType;
}

std::string Entity::show() {
    std::string message = ModelElement::show();
    if (this->_entityType != nullptr) {
        message += ",entityType=\"" + this->_entityType->getName() + "\"";
    }
    message += ",attributes=[";
    _attributeValues->front();
    for (unsigned int i = 0; i < _attributeValues->size(); i++) {
        std::map<std::string, double>* map = _attributeValues->current();
        std::string attributeName = _parentModel->getElements()->getElementList(Util::TypeOf<Attribute>())->getAtRank(i)->getName();
        message += attributeName + "=";
        if (map->size() == 0) { // scalar
            message += "NaN;"; //std::to_string(map->begin()->second) + ";";
        } else if (map->size() == 1) { // scalar
            message += strTruncIfInt(std::to_string(map->begin()->second)) + ", ";
        } else {
            // array or matrix
            message += "[";
            for (std::pair<std::string, double> valIt : *map) {
                message += valIt.first + "=>" + strTruncIfInt(std::to_string(valIt.second)) + ", ";
            }
            message = message.substr(0, message.length() - 2);
            message += "];";
        }
        _attributeValues->next();
    }
    message = message.substr(0, message.length() - 1);
    message += "]";

    return message;
}

double Entity::getAttributeValue(std::string attributeName) {

    return getAttributeValue("", attributeName);
}

double Entity::getAttributeValue(std::string index, std::string attributeName) {
    int rank = _parentModel->getElements()->getRankOf(Util::TypeOf<Attribute>(), attributeName);
    if (rank >= 0) {
        std::map<std::string, double>* map = this->_attributeValues->getAtRank(rank);
        std::map<std::string, double>::iterator mapIt = map->find(index);
        if (mapIt != map->end()) {//found
            return (*mapIt).second;
        } else { // not found
            return 0.0;
        }
    }
    _parentModel->getTracer()->trace(Util::TraceLevel::L3_errorRecover, "Attribute \"" + attributeName + "\" not found");

    return 0.0; /* \todo: !! Never should happen. check how to report */
}

double Entity::getAttributeValue(Util::identification attributeID) {

    return getAttributeValue("", attributeID);
}

double Entity::getAttributeValue(std::string index, Util::identification attributeID) {
    //assert(this->_parentModel != nullptr);
    ModelElement* element = _parentModel->getElements()->getElement(Util::TypeOf<Attribute>(), attributeID);
    if (element != nullptr) {

        return getAttributeValue(index, element->getName());
    }
    return 0.0; // attribute not found
}

void Entity::setAttributeValue(std::string attributeName, double value) {
    setAttributeValue("", attributeName, value);
}

void Entity::setAttributeValue(std::string index, std::string attributeName, double value) {
    int rank = _parentModel->getElements()->getRankOf(Util::TypeOf<Attribute>(), attributeName);
    if (rank >= 0) {
        std::map<std::string, double>* map = _attributeValues->getAtRank(rank);
        std::map<std::string, double>::iterator mapIt = map->find(index);
        if (mapIt != map->end()) {//found
```

```
            (*mapIt).second = value;
        } else { // not found
            map->insert({index, value}); // (map->end(), std::pair<std::string, double>(index, value));
        }
    } else
        _parentModel->getTracer()->trace(Util::TraceLevel::L3_errorRecover, "Attribute \"" + attributeName + "\" not found");

}

void Entity::setAttributeValue(Util::identification attributeID, double value) {

    setAttributeValue("", attributeID, value);
}

void Entity::setAttributeValue(std::string index, Util::identification attributeID, double value) {

    std::string attrname = _parentModel->getElements()->getElement(Util::TypeOf<Attribute>(), attributeID)->getName();
    setAttributeValue(index, attrname, value);
}

Util::identification Entity::entityNumber() const {

    return _entityNumber;
}

bool Entity::_loadInstance(std::map<std::string, std::string>* fields) {
    // never loads an entity
    fields = nullptr; // just to use the parameter and avoid warning for not using it

    return fields;
}

std::map<std::string, std::string>* Entity::_saveInstance() {

    return new std::map<std::string, std::string>();
}

bool Entity::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}
```

8.165 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Entity.h File Reference

```
#include <string>
#include <map>
#include "../util/Util.h"
#include "../util/List.h"
#include "ModelElement.h"
#include "EntityType.h"
```

Include dependency graph for Entity.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Entity

8.166 Entity.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Entity.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 16:30
 */

#ifndef ENTITY_H
#define ENTITY_H

#include <string>
#include <map>

#include "../util/Util.h"
#include "../util/List.h"
#include "ModelElement.h"
#include "EntityType.h"
//namespace GenesysKernel {

class Entity : public ModelElement {
private: // no one can create or destry entities directlly. This can be done one throught friend class Model
    Entity(Model* model, std::string name = "", bool insertIntoModel = true);
    virtual ~Entity() = default;
    // friend Entity* Model::createEntity(std::string name, bool insertIntoModel); // It would be better, but Model is not known at this point of compilaton
    friend class Model;
public:
    virtual std::string show();

public: // g & s
    void setEntityTypeName(std::string entityTypeName) throw (); //*!< indirect access to EntityType
    std::string getEntityTypeName() const;
    void setEntityType(EntityType* entityType); //*!< direct access to EntityType
    EntityType* getEntityType() const;
public:
    double getAttributeValue(std::string attributeName);
    double getAttributeValue(std::string index, std::string attributeName);
    double getAttributeValue(Util::identification attributeID);
    double getAttributeValue(std::string index, Util::identification attributeID);
    void setAttributeValue(std::string attributeName, double value);
    void setAttributeValue(std::string index, std::string attributeName, double value);
    void setAttributeValue(Util::identification attributeID, double value);
    void setAttributeValue(std::string index, Util::identification attributeID, double value);
    Util::identification entityNumber() const;
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:
    Util::identification _entityNumber;
    EntityType* _entityType = nullptr;
```

```
    List< std::map<std::string, double>* >* _attributeValues = new List<std::map<std::string, double>*>();
};
//namespace\\}
#endif /* ENTITY_H */
```

8.167 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/EntityType.cpp File Reference

```
#include "EntityType.h"
#include "Model.h"
#include "SimulationResponse.h"
```

Include dependency graph for EntityType.cpp:

8.168 EntityType.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   EntityType.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 19:24
 */

#include "EntityType.h"
#include "Model.h"
#include "SimulationResponse.h"

//using namespace GenesysKernel;

EntityType::EntityType(Model* model, std::string name) : ModelElement(model, Util::TypeOf<EntityType>(), name) {
}

void EntityType::_initBetweenReplications() {
    _initialWaitingCost = 0.0;
    _initialVACost = 0.0;
    _initialNVACost = 0.0;
    _initialOtherCost = 0.0;
    for (StatisticsCollector* cstat : *_statisticsCollectors->list()) {
        cstat->getStatistics()->getCollector()->clear();
    }
}

EntityType::~EntityType() {
```

```
    // remove all CStats
    for (StatisticsCollector* cstat : *_statisticsCollectors->list()) {
        _parentModel->getElements()->remove(Util::TypeOf<StatisticsCollector>(), cstat);
    }
}

std::string EntityType::show() {
    return ModelElement::show() +
            ",initialPicture=" + this->_initialPicture; // add more...
}

void EntityType::setInitialWaitingCost(double _initialWaitingCost) {
    this->_initialWaitingCost = _initialWaitingCost;
}

double EntityType::initialWaitingCost() const {
    return _initialWaitingCost;
}

void EntityType::setInitialOtherCost(double _initialOtherCost) {
    this->_initialOtherCost = _initialOtherCost;
}

double EntityType::initialOtherCost() const {
    return _initialOtherCost;
}

void EntityType::setInitialNVACost(double _initialNVACost) {
    this->_initialNVACost = _initialNVACost;
}

double EntityType::initialNVACost() const {
    return _initialNVACost;
}

void EntityType::setInitialVACost(double _initialVACost) {
    this->_initialVACost = _initialVACost;
}

double EntityType::initialVACost() const {
    return _initialVACost;
}

void EntityType::setInitialPicture(std::string _initialPicture) {
    this->_initialPicture = _initialPicture;
}

std::string EntityType::initialPicture() const {
    return _initialPicture;
}

StatisticsCollector* EntityType::addGetStatisticsCollector(std::string name) {
    for (StatisticsCollector* cstat : *_statisticsCollectors->list()) {
        if (cstat->getName() == name) {
            return cstat;
        }
    }
    // not found. Create it, insert it into the list of cstats, into the model element manager, and then return it
    StatisticsCollector* cstat = new StatisticsCollector(_parentModel, name, this);
    _statisticsCollectors->insert(cstat); // \todo _statisticsCollectors list is probabily redundant to _childrenElements and unnecessary
    _childrenElements->insert({name, cstat});
    //_parentModel->insert(cstat); // unnecessary
    return cstat;
}

PluginInformation* EntityType::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<EntityType>(), &EntityType::LoadInstance);
    return info;
}

ModelElement* EntityType::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    EntityType* newElement = new EntityType(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool EntityType::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        this->_initialNVACost = LoadField(fields, "initialNVACost", DEFAULT.initialCost);
```

```
        this->_initialOtherCost = LoadField(fields, "initialOtherCost", DEFAULT.initialCost);
        this->_initialVACost = LoadField(fields, "initialVACost", DEFAULT.initialCost);
        this->_initialWaitingCost = LoadField(fields, "initialWaitingCost", DEFAULT.initialCost);
        this->_initialPicture = LoadField(fields, "initialPicture", DEFAULT.initialPicture);
    }
    return res;
}

std::map<std::string, std::string>* EntityType::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<EntityType>());
    SaveField(fields, "initialNVACost", _initialNVACost, DEFAULT.initialCost);
    SaveField(fields, "initialOtherCost", _initialOtherCost, DEFAULT.initialCost);
    SaveField(fields, "initialVACost", _initialVACost, DEFAULT.initialCost);
    SaveField(fields, "initialWaitingCost", _initialWaitingCost, DEFAULT.initialCost);
    SaveField(fields, "initialPicture", _initialPicture, DEFAULT.initialPicture);
    return fields;
}

bool EntityType::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}

void EntityType::_createInternalElements() {
    if (_reportStatistics) {
    } else {
        while (_statisticsCollectors->size() > 0) {
            _parentModel->getElements()->remove(_statisticsCollectors->front());
            _statisticsCollectors->front()->~StatisticsCollector();
            _statisticsCollectors->pop_front();
        }
    }
}

```

8.169 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/EntityType.h File Reference

```
#include <string>
#include "ModelElement.h"
#include "StatisticsCollector.h"
#include "ElementManager.h"
#include "Plugin.h"
```

Include dependency graph for EntityType.h:

Classes

- class EntityType

8.170 EntityType.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   EntityType.h
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 19:24
 */

#ifndef ENTITYTYPE_H
#define ENTITYTYPE_H

#include <string>
#include "ModelElement.h"
#include "StatisticsCollector.h"
#include "ElementManager.h"
#include "Plugin.h"

//#include "Model.h"
//namespace GenesysKernel {

class EntityType : public ModelElement {
public:
    EntityType(Model* model, std::string name = "");
    virtual ~EntityType();
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public: //get & set
    void setInitialWaitingCost(double _initialWaitingCost);
    double initialWaitingCost() const;
    void setInitialOtherCost(double _initialOtherCost);
    double initialOtherCost() const;
    void setInitialNVACost(double _initialNVACost);
    double initialNVACost() const;
    void setInitialVACost(double _initialVACost);
    double initialVACost() const;
    void setInitialPicture(std::string _initialPicture);
    std::string initialPicture() const;
public: //get
    StatisticsCollector* addGetStatisticsCollector(std::string name);

protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _initBetweenReplications();
    virtual void _createInternalElements();
private:
    void _initCostsAndStatistics();
private:

    const struct DEFAULT_VALUES {
        std::string initialPicture = "report";
        double initialCost = 0.0;
    } DEFAULT;
    std::string _initialPicture = DEFAULT.initialPicture;
    double _initialVACost = DEFAULT.initialCost;
    double _initialNVACost = DEFAULT.initialCost;
    double _initialOtherCost = DEFAULT.initialCost;
    double _initialWaitingCost = DEFAULT.initialCost;
private: //1:n
    List<StatisticsCollector*>* _statisticsCollectors = new List<StatisticsCollector*>();
};
//namespace\\}
#endif /* ENTITYTYPE_H */

```

8.171 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Event.cpp File Reference

```
#include "Event.h"
```

Include dependency graph for Event.cpp:

8.172 Event.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Event.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:41
 */

#include "Event.h"

//using namespace GenesysKernel;

Event::Event(double time, Entity* entity, ModelComponent* component, unsigned int componentInputNumber) {
    _time = time;
    _entity = entity;
    _component = component;
    _componentInputNumber = componentInputNumber;
}

Event::Event(double time, Entity* entity, Connection* connection) {
    _time = time;
    _entity = entity;
    _component = connection->first;
    _componentInputNumber = connection->second;
}

std::string Event::show() {
    std::string message = "time=" + std::to_string(_time) + //Util::StrTimeUnit(???)+
            ",entity=" + _entity->getName() + //std::to_string(_entity->entityNumber()) +
            ",component=\"" + _component->getName() + "\"";
 //+std::to_string(_component->getId())+"}";
    if (this->_componentInputNumber > 0) {
        message += ",input=" + std::to_string(this->_componentInputNumber);
    }
    return message;
}

unsigned int Event::getComponentInputNumber() const {
    return _componentInputNumber;
}

double Event::getTime() const {
    return _time;
}

ModelComponent* Event::getComponent() const {
    return _component;
}

Entity* Event::getEntity() const {
```

```
    return _entity;
}
```

8.173 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Event.h File Reference

```
#include <string>
#include "ModelElement.h"
#include "Entity.h"
#include "ModelComponent.h"
```

Include dependency graph for Event.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Event

8.174 Event.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Event.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:41
 */

#ifndef EVENT_H
#define EVENT_H

#include <string>

#include "ModelElement.h"
#include "Entity.h"
```

```
#include "ModelComponent.h"

//namespace GenesysKernel {

class Event {//: public ModelElement {
public:
    Event(double time, Entity* entity, ModelComponent* component, unsigned int componentInputNumber = 0);
    Event(double time, Entity* entity, Connection* connection);
    virtual ~Event() = default;
public:
    double getTime() const;
    ModelComponent* getComponent() const;
    Entity* getEntity() const;
    unsigned int getComponentInputNumber() const;
public:
    std::string show();
private:
    double _time;
    Entity* _entity;
    ModelComponent* _component;
    unsigned int _componentInputNumber;
};
//namespace\\}
#endif /* EVENT_H */

```

8.175 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Experiment↩ Manager.cpp File Reference

```
#include "ExperimentManager.h"
#include "Simulator.h"
```

Include dependency graph for ExperimentManager.cpp:

8.176 ExperimentManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ExperimentManager.cpp
 * Author: rlcancian
 *
 * Created on 4 de maio de 2021, 11:43
 */

#include "ExperimentManager.h"
```

```
#include "Simulator.h"

ExperimentManager::ExperimentManager(Simulator* simulator) {
    _simulator = simulator;
    _currentSimulationExperiment = nullptr;
}

List<SimulationExperiment*>* ExperimentManager::getExperiments() const {
    return _experiments;
}

SimulationExperiment* ExperimentManager::newSimulationExperiment() {
    _currentSimulationExperiment = new SimulationExperiment(); //_simulator
    return _currentSimulationExperiment;
}

void ExperimentManager::insert(SimulationExperiment* experiment) {
    _experiments->insert(experiment);
    this->_currentSimulationExperiment = experiment;
    _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Experiment successfully inserted");
}

void ExperimentManager::remove(SimulationExperiment* experiment) {
    _experiments->remove(experiment);
    if (_currentSimulationExperiment == experiment) {
        _currentSimulationExperiment = this->front();
    }
    experiment->~SimulationExperiment();
    _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Experiment successfully removed");
}

unsigned int ExperimentManager::size() {
    return _experiments->size();
}

bool ExperimentManager::saveSimulationExperiment(std::string filename) {
    // \todo: implement
    if (_currentSimulationExperiment != nullptr) {
        filename = ""; // just to avoid not using it
        //return _currentSimulationExperiment->save(filename);
    }
    return false;
}

bool ExperimentManager::loadSimulationExperiment(std::string filename) {
    SimulationExperiment* experiment = new SimulationExperiment(); //_simulator);
    // \todo: not implemented yet!
    filename = ""; // jut to avoid not using it
    bool res = false; // = experiment->load(filename);
    if (res) {
        this->insert(experiment);
        _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Experiment successfully loaded");
    } else {
        experiment->~SimulationExperiment();
        _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Experiment coud not be loaded");
    }
    return res;
}

void ExperimentManager::setCurrent(SimulationExperiment* experiment) {
    this->_currentSimulationExperiment = experiment;
}

SimulationExperiment* ExperimentManager::current() {
    return _currentSimulationExperiment;
}

SimulationExperiment* ExperimentManager::front() {
    return _experiments->front();
}

SimulationExperiment* ExperimentManager::next() {
    return _experiments->next();
}

```

8.177 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ExperimentManager.h File Reference

```
#include "PersistentObject_base.h"
```

```
#include "SimulationExperiment.h"
#include "../util/List.h"
```

Include dependency graph for ExperimentManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ExperimentManager

8.178 ExperimentManager.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ExperimentManager.h
00009  * Author: rlcancian
00010  *
00011  * Created on 4 de maio de 2021, 11:43
00012  */
00013
00014 #ifndef EXPERIMENTMANAGER_H
00015 #define EXPERIMENTMANAGER_H
00016
00017 //#include "Simulator.h"
00018 #include "PersistentObject_base.h"
00019 //#include "SimulationControl.h"
00020 //#include "SimulationResponse.h"
00021 #include "SimulationExperiment.h"
00022 #include "../util/List.h"
00023
00024 class Simulator;
00025
00026 class ExperimentManager {// : PersistentObject_base {
00027 public:
00028
00029     /*  class ModelSimulationInsOuts {
00030         public:
00031             ModelSimulationInsOuts() = default;
00032             struct ControlResponseInfos {
00033                 std::string type;
00034                 std::string parent;
00035                 std::string name;
00036                 double defaultValue;
00037             };
```

```
            List<ControlResponseInfos*>* getResponsesInfos() const {
                return _responsesInfos;
            }

            std::string getModelfilename() const {
                return _experimentfilename;
            }

            List<ControlResponseInfos*>* getControlInfos() const {
                return _controlInfos;
            }
        private:
            std::string _modelfilename;
            List<ControlResponseInfos*>* _responsesInfos = new List<ControlResponseInfos*>();
            List<ControlResponseInfos*>* _controlInfos = new List<ControlResponseInfos*>();
        }; */

public:
    ExperimentManager(Simulator* simulator);

public:
    SimulationExperiment* newSimulationExperiment();
    void insert(SimulationExperiment* experiment);
    void remove(SimulationExperiment* experiment);
    void setCurrent(SimulationExperiment* experiment);
    bool saveSimulationExperiment(std::string filename);
    bool loadSimulationExperiment(std::string filename);
    unsigned int size();
public:
    SimulationExperiment* front();
    SimulationExperiment* current();
    SimulationExperiment* next();
    //SimulationExperiment* end();
public:
    List<SimulationExperiment*>* getExperiments() const;
private:
    Simulator* _simulator;
    SimulationExperiment* _currentSimulationExperiment;
    List<SimulationExperiment*>* _experiments = new List<SimulationExperiment*>();

};

#endif /* EXPERIMENTMANAGER_H */

```

8.179 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.cpp File Reference

```
#include "ExperimentManagerDefaultImpl1.h"
```

Include dependency graph for ExperimentManagerDefaultImpl1.cpp:

8.180 ExperimentManagerDefaultImpl1.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ProcessAnalyserDefaultImpl1.cpp
 * Author: rlcancian
 *
 * Created on 20 de Maio de 2019, 20:45
 */

#include "ExperimentManagerDefaultImpl1.h"

ExperimentManagerDefaultImpl1::ExperimentManagerDefaultImpl1() {
}

List<SimulationScenario*>* ExperimentManagerDefaultImpl1::getScenarios() const {
    // \todo: implement
    return nullptr;
}

List<SimulationControl*>* ExperimentManagerDefaultImpl1::getControls() const {
    return _controls;
}

List<SimulationResponse*>* ExperimentManagerDefaultImpl1::getResponses() const {
    // \todo: implement
    return nullptr;
}

List<SimulationControl*>* ExperimentManagerDefaultImpl1::extractControlsFromModel(std::string modelFilename) const {
    // \todo: implement
    modelFilename = ""; //just to use it
    return nullptr;
}

List<SimulationResponse*>* ExperimentManagerDefaultImpl1::extractResponsesFromModel(std::string modelFilename) const {
    // \todo: implement
    modelFilename = ""; // juts to use it
    return nullptr;
}
```

```
00044
00045 void ExperimentManagerDefaultImpl1::startSimulationOfScenario(SimulationScenario* scenario) {
00046     // \todo: implement
00047 }
00048
00049 void ExperimentManagerDefaultImpl1::startExperiment() {
00050     // \todo: implement
00051 }
00052
00053 void ExperimentManagerDefaultImpl1::stopExperiment() {
00054     // \todo: implement
00055 }
00056
00057 void ExperimentManagerDefaultImpl1::addTraceSimulationHandler(traceSimulationProcessListener
       traceSimulationProcessListener) {
00058 }
```

8.181 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ExperimentManagerDefaultImpl1.h File Reference

```
#include "ExperimetManager_if.h"
#include "SimulationScenario.h"
#include "SimulationResponse.h"
#include "SimulationControl.h"
```

Include dependency graph for ExperimentManagerDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ExperimentManagerDefaultImpl1

8.182 ExperimentManagerDefaultImpl1.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ProcessAnalyserDefaultImpl1.h
 * Author: rlcancian
 *
 * Created on 20 de Maio de 2019, 20:45
 */

#ifndef PROCESSANALYSERDEFAULTIMPL1_H
#define PROCESSANALYSERDEFAULTIMPL1_H

#include "ExperimetManager_if.h"
#include "SimulationScenario.h"
#include "SimulationResponse.h"
#include "SimulationControl.h"

class ExperimentManagerDefaultImpl1 : public ExperimentManager_if {
public:
    ExperimentManagerDefaultImpl1();
    virtual ~ExperimentManagerDefaultImpl1() = default;
public:
    virtual List<SimulationScenario*>* getScenarios() const;
    virtual List<SimulationControl*>* getControls() const;
    virtual List<SimulationResponse*>* getResponses() const;
    virtual List<SimulationControl*>* extractControlsFromModel(std::string modelFilename) const;
    virtual List<SimulationResponse*>* extractResponsesFromModel(std::string modelFilename) const;
    virtual void startSimulationOfScenario(SimulationScenario* scenario);
    virtual void startExperiment();
    virtual void stopExperiment();
    virtual void addTraceSimulationHandler(traceSimulationProcessListener
 traceSimulationProcessListener);
private:
    List<SimulationControl*>* _controls = new List<SimulationControl*>();
};

#endif /* PROCESSANALYSERDEFAULTIMPL1_H */
```

8.183 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ExperimetManager_↩ if.h File Reference

```
#include "../util/List.h"
#include "SimulationScenario.h"
#include "SimulationControl.h"
#include "SimulationResponse.h"
#include "TraceManager.h"
```

Include dependency graph for ExperimetManager_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ExperimentManager_if

8.184 ExperimetManager_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ExperimentManager_if (old ProcessAnalyser_if.h)
 * Author: rafael.luiz.cancian
 *
 * Created on 10 de Outubro de 2018, 14:26
 */

#ifndef EXPERIMENTMANAGER_IF_H
#define EXPERIMENTMANAGER_IF_H

#include "../util/List.h"
#include "SimulationScenario.h"
#include "SimulationControl.h"
#include "SimulationResponse.h"
#include "TraceManager.h"

class ExperimentManager_if {
public:
    virtual List<SimulationScenario*>* getScenarios() const = 0;
    virtual List<SimulationControl*>* getControls() const = 0;
    virtual List<SimulationResponse*>* getResponses() const = 0;
    virtual List<SimulationControl*>* extractControlsFromModel(std::string modelFilename) const = 0;
    virtual List<SimulationResponse*>* extractResponsesFromModel(std::string modelFilename) const = 0;
    virtual void startSimulationOfScenario(SimulationScenario* scenario) = 0;
    virtual void startExperiment() = 0;
    virtual void stopExperiment() = 0;
    virtual void addTraceSimulationHandler(traceSimulationProcessListener
 traceSimulationProcessListener) = 0;
};

#endif /* EXPERIMENTMANAGER_IF_H */

```

8.185 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/LicenceManager.cpp File Reference

```
#include "LicenceManager.h"
```

Include dependency graph for LicenceManager.cpp:

8.186 LicenceManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   LicenceManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 29 de Maio de 2019, 11:45
 */

#include "LicenceManager.h"

//using namespace GenesysKernel;

LicenceManager::LicenceManager(Simulator* simulator) {
    _simulator = simulator;
    this->setDefaultLicenceAndLimits();
}

void LicenceManager::setDefaultLicenceAndLimits() {
    _licence = "LICENCE: Academic Mode. In academic mode this software has full functionality and executing training-size simulation models. This software may be duplicated and used for educational purposes only; any commercial application is a violation of the license agreement. Designed and developed by prof. Dr. Ing Rafael Luiz Cancian, 2018-2021";
    _activationCode = "";
    _components = 50;
    _elements = 100;
    _entities = 200;
    _hosts = 1;
    _threads = 1;
}

const std::string LicenceManager::showLicence() const {
    return _licence;
}

const std::string LicenceManager::showLimits() const {
    std::string msg = "LIMITS: Based on your licence and activation code, your simulator is running under the following limits" +
            std::string(": ") + std::to_string(_components) + " components" +
            ", " + std::to_string(_elements) + " elements" +
            ", " + std::to_string(_entities) + " entities" +
            ", " + std::to_string(_hosts) + " hosts" +
            ", " + std::to_string(_threads) + " threads.";
    return msg;
}
```

```
00046
00047 const std::string LicenceManager::showActivationCode() const {
00048     std::string msg = "ACTIVATION CODE: Not found.";
00049     return msg;
00050 }
00051
00052 bool LicenceManager::lookforActivationCode() {
00053     // \todo: Not implemented yet
00054     return false;
00055 }
00056
00057 bool LicenceManager::insertActivationCode() {
00058     // \todo: Not implemented yet
00059     return false;
00060 }
00061
00062 void LicenceManager::removeActivationCode() {
00063     this->setDefaultLicenceAndLimits();
00064 }
00065
00066 unsigned int LicenceManager::getModelComponentsLimit() {
00067     return this->_components;
00068 }
00069
00070 unsigned int LicenceManager::getModelElementsLimit() {
00071     return this->_elements;
00072 }
00073
00074 unsigned int LicenceManager::getEntityLimit() {
00075     return this->_entities;
00076 }
00077
00078 unsigned int LicenceManager::getHostsLimit() {
00079     return this->_hosts;
00080 }
00081
00082 unsigned int LicenceManager::getThreadsLimit() {
00083     return this->_threads;
00084 }
```

8.187 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/LicenceManager.h File Reference

`#include <string>`

Include dependency graph for LicenceManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class LicenceManager

8.188 LicenceManager.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   LicenceManager.h
 * Author: rafael.luiz.cancian
 *
 * Created on 29 de Maio de 2019, 11:45
 */

#ifndef LICENCEMANAGER_H
#define LICENCEMANAGER_H

#include <string>

//namespace GenesysKernel {
class Simulator;

class LicenceManager {
public:
    LicenceManager(Simulator* simulator);
    virtual ~LicenceManager() = default;
    const std::string showLicence() const;
    const std::string showLimits() const;
    const std::string showActivationCode() const;
    bool lookforActivationCode();
    bool insertActivationCode();
    void removeActivationCode();
    unsigned int getModelComponentsLimit();
    unsigned int getModelElementsLimit();
    unsigned int getEntityLimit();
    unsigned int getHostsLimit();
    unsigned int getThreadsLimit();
private:
    void setDefaultLicenceAndLimits();
private:
    Simulator* _simulator;
    std::string _licence;
    std::string _activationCode;
    unsigned int _components, _elements, _entities, _hosts, _threads;
};
//namespace\\}
#endif /* LICENCEMANAGER_H */

```

8.189 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Model.cpp File Reference

```
#include <typeinfo>
#include <iostream>
#include <algorithm>
#include <string>
#include "Model.h"
#include "SourceModelComponent.h"
#include "Simulator.h"
#include "OnEventManager.h"
#include "StatisticsCollector.h"
```

```
#include "../TraitsKernel.h"
```

Include dependency graph for Model.cpp:

Functions

- bool EventCompare (const Event ∗a, const Event ∗b)

8.189.1 Function Documentation

8.189.1.1 EventCompare()

```
bool EventCompare (
    const Event * a,
    const Event * b )
```

Definition at line 29 of file Model.cpp.

Here is the call graph for this function:

8.190 Model.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   SimulationModel.cpp
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 15:01
```

```
 */

#include <typeinfo>
#include <iostream>
#include <algorithm>
#include <string>

#include "Model.h"
#include "SourceModelComponent.h"
#include "Simulator.h"
#include "OnEventManager.h"
#include "StatisticsCollector.h"
#include "../TraitsKernel.h"
//#include "Access.h"

//using namespace GenesysKernel;

bool EventCompare(const Event* a, const Event * b) {
    return a->getTime() < b->getTime();
}

Model::Model(Simulator* simulator) {
    _parentSimulator = simulator; // a simulator is the "parent" of a model
    // 1:1 associations (no Traits)
    _traceManager = simulator->getTracer(); // every model starts with the same tracer, unless a specific one is set
    _modelInfo = new ModelInfo();
    _eventManager = new OnEventManager(); // should be on .h (all that does not depends on THIS)
    _elementManager = new ElementManager(this);
    _componentManager = new ComponentManager(this);
    _simulation = new ModelSimulation(this);
    // 1:1 associations (Traits)
    //Sampler_if* sampler = new Traits<Sampler_if>::Implementation();
    _parser = new TraitsKernel<Parser_if>::Implementation(this, new TraitsKernel<Sampler_if>::Implementation());
    _modelChecker = new TraitsKernel<ModelChecker_if>::Implementation(this);
    _modelPersistence = new TraitsKernel<ModelPersistence_if>::Implementation(this);
    // 1:n associations
    _futureEvents = new List<Event*>();
    //_events->setSortFunc(&EventCompare); // It works too
    _futureEvents->setSortFunc([](const Event* a, const Event * b) {
        return a->getTime() < b->getTime();
    });
    // for process analyser
    _responses = new List<SimulationResponse*>();
    _controls = new List<SimulationControl*>();
    // insert controls
    _controls->insert(new SimulationControl("ModelSimulation", "NumberOfReplications",
            DefineGetterMember<ModelSimulation>(this->_simulation, &ModelSimulation::getNumberOfReplications),
            DefineSetterMember<ModelSimulation>(this->_simulation, &ModelSimulation::setNumberOfReplications))
            );
    _controls->insert(new SimulationControl("ModelSimulation", "ReplicationLength",
            DefineGetterMember<ModelSimulation>(this->_simulation, &ModelSimulation::getReplicationLength),
            DefineSetterMember<ModelSimulation>(this->_simulation, &ModelSimulation::setReplicationLength))
            );
    _controls->insert(new SimulationControl("ModelSimulation", "WarmupPeriod",
            DefineGetterMember<ModelSimulation>(this->_simulation, &ModelSimulation::getWarmUpPeriod),
            DefineSetterMember<ModelSimulation>(this->_simulation, &ModelSimulation::setWarmUpPeriod))
            );
}

void Model::sendEntityToComponent(Entity* entity, Connection* connection, double timeDelay) {
    this->sendEntityToComponent(entity, connection->first, timeDelay, connection->second);
}

void Model::sendEntityToComponent(Entity* entity, ModelComponent* component, double timeDelay, unsigned int componentInputNumber) {
    SimulationEvent* se = _simulation->_createSimulationEvent();
    se->setDestinationComponent(component);
    this->getOnEvents()->NotifyEntityMoveHandlers(se); // it's my friend  //\todo: Event should include information about "from component" and timeDelay, but it doesn't
    // schedule to send it
    Event* newEvent = new Event(this->getSimulation()->getSimulatedTime() + timeDelay, entity, component, componentInputNumber);
    this->getFutureEvents()->insert(newEvent);
}

bool Model::save(std::string filename) {
    bool res = this->_modelPersistence->save(filename);
    if (res) {
        this->_traceManager->trace(Util::TraceLevel::L2_results, "Model successfully saved");
    } else {
        this->_traceManager->trace(Util::TraceLevel::L2_results, "Model could not be saved");
```

```

    }
    return res;
}

bool Model::load(std::string filename) {
    this->clear();
    bool res = this->_modelPersistence->load(filename);
    if (res)
        this->_traceManager->trace(Util::TraceLevel::L2_results, "Model successfully loaded");
    else
        this->_traceManager->trace(Util::TraceLevel::L2_results, "Model could not be loaded");
    return res;
}

double Model::parseExpression(const std::string expression) {
    try {
        return _parser->parse(expression);
    } catch (...) {
        return 0.0; // \todo: HOW SAY THERE WAS AN ERROR?
    }
}

bool Model::checkExpression(const std::string expression, const std::string expressionName, std::string* errorMessage) {
    bool result;
    getTracer()->trace("Checking expression \"" + expression + "\"", Util::TraceLevel::L8_detailed);
    try {
        parseExpression(expression, &result, errorMessage);
    } catch (...) {
        result = false;
    }
    if (!result) {
        std::string msg = "Expression \"" + expression + "\" for '" + expressionName + "' is incorrect. ";
        this->_traceManager->trace(msg, Util::TraceLevel::L3_errorRecover);
        errorMessage->append(msg);
    }
    return result;
}

double Model::parseExpression(const std::string expression, bool* success, std::string* errorMessage) {
    double value = _parser->parse(expression, success, errorMessage);
    return value;
}

void Model::show() {
    getTracer()->trace(Util::TraceLevel::L2_results, "Simulation Model:");
    Util::IncIndent();
    {
        getTracer()->trace(Util::TraceLevel::L2_results, "Information:");
        Util::IncIndent();
        getTracer()->trace(Util::TraceLevel::L2_results, this->getInfos()->show());
        Util::DecIndent();
        _showConnections();
        _showComponents();
        _showElements();
        _showSimulationControls();
        _showSimulationResponses();
    }
    Util::DecIndent();
    getTracer()->trace(Util::TraceLevel::L2_results, "End of Simulation Model");
}

bool Model::insert(ModelElement* elemOrComp) {
    ModelComponent* comp = dynamic_cast<ModelComponent*> (elemOrComp);
    if (comp == nullptr) // it's a ModelElement
        return this->getElements()->insert(elemOrComp);
    else // it's a ModelComponent
        return this->getComponents()->insert(comp);
}

void Model::remove(ModelElement* elemOrComp) {
    ModelComponent* comp = dynamic_cast<ModelComponent*> (elemOrComp);
    if (comp == nullptr) // it's a ModelElement
        this->getElements()->remove(elemOrComp);
    else // it's a ModelComponent
        this->getComponents()->remove(comp);
}

void Model::_showElements() const {
    getTracer()->trace(Util::TraceLevel::L2_results, "Elements:");
    Util::IncIndent();
    {
        std::string elementType;
        ModelElement* element;
```

```
        std::list<std::string>* elementTypes = getElements()->getElementClassnames();
        for (std::list<std::string>::iterator typeIt = elementTypes->begin(); typeIt != elementTypes->end(); typeIt++) {
            elementType = (*typeIt);
            List<ModelElement*>* em = getElements()->getElementList(elementType);
            getTracer()->trace(Util::TraceLevel::L2_results, elementType + ":");
            Util::IncIndent();
            {
                for (std::list<ModelElement*>::iterator it = em->list()->begin(); it != em->list()->end(); it++) {
                    element = (*it);
                    getTracer()->trace(Util::TraceLevel::L2_results, element->show());
                }
            }
            Util::DecIndent();
        }
    }
    Util::DecIndent();
}

void Model::_showConnections() const {
    // \todo
}

void Model::_showComponents() const {
    getTracer()->trace(Util::TraceLevel::L2_results, "Components:");
    Util::IncIndent();
    for (std::list<ModelComponent*>::iterator it = getComponents()->begin(); it != getComponents()->end(); it++) {
        getTracer()->trace(Util::TraceLevel::L2_results, (*it)->show());
    }
    Util::DecIndent();
}

void Model::_showSimulationControls() const {
    getTracer()->trace(Util::TraceLevel::L2_results, "Simulation Controls:");
    Util::IncIndent();
    for (std::list<SimulationControl*>::iterator it = _controls->list()->begin(); it != _controls->list()->end(); it++) {
        getTracer()->trace(Util::TraceLevel::L2_results, (*it)->show());
    }
    Util::DecIndent();
}

void Model::_showSimulationResponses() const {
    getTracer()->trace(Util::TraceLevel::L2_results, "Simulation Responses:");
    Util::IncIndent();
    for (std::list<SimulationResponse*>::iterator it = _responses->list()->begin(); it != _responses->list()->end(); it++) {
        getTracer()->trace(Util::TraceLevel::L2_results, (*it)->show());
    }
    Util::DecIndent();
}

void Model::clear() {
    this->_componentManager->clear();
    this->_elementManager->clear();
    this->_futureEvents->clear();
    //this->_simulation->clear();  // \todo clear method
    //this->_modelInfo->clear(); // \todo clear method
    //Util::ResetAllIds(); // \todo: To implement
}

void Model::_createModelInternalElements() {
    getTracer()->trace(Util::TraceLevel::L7_internal, "Creating internal elements");
    Util::IncIndent();

    for (ModelComponent* component : *_componentManager) {
        getTracer()->trace(Util::TraceLevel::L8_detailed, "Internals for " + component->getClassname() + " \"" + component->getName() + "\"");
        Util::IncIndent();
        ModelComponent::CreateInternalElements(component);
        Util::DecIndent();
    }

    std::list<ModelElement*>* modelElements;
    unsigned int originalSize = getElements()->getElementClassnames()->size(), pos = 1;
    //for (std::list<std::string>::iterator itty = elements()->elementClassnames()->begin(); itty != elements()->elementClassnames()->end(); itty++) {
    std::list<std::string>::iterator itty = getElements()->getElementClassnames()->begin();
    while (itty != getElements()->getElementClassnames()->end() && pos <= originalSize) {
        //try {
        modelElements = getElements()->getElementList((*itty))->list();
        //} catch (const std::exception& e) {
        // \todo Is there a better solution to iterate over a changing sorted list??
        // ops. Sorted list has changed and iteration fails. Starts iterating again
        //  itty = elements()->elementClassnames()->begin();
```

```
        //  modelElements = elements()->elementList((*itty))->list();
        //  tracer()->trace(Util::TraceLevel::L7_internal, "Creating internal elements");
        //}
        for (std::list<ModelElement*>::iterator itel = modelElements->begin(); itel != modelElements->end(); itel++) {
            getTracer()->trace(Util::TraceLevel::L8_detailed, "Internals for " + (*itel)->getClassname() + " \"" + (*itel)->getName() + "\""); // (" + std::to_string(pos) + "/" + std::to_string(originalSize) + ")");
            Util::IncIndent();
            ModelElement::CreateInternalElements((*itel));
            Util::DecIndent();
        }
        if (originalSize == getElements()->getElementClassnames()->size()) {
            itty++;
            pos++;
        } else {
            originalSize = getElements()->getElementClassnames()->size();
            itty = getElements()->getElementClassnames()->begin();
            pos = 1;
            getTracer()->trace(Util::TraceLevel::L7_internal, "Restarting to create internal elements (due to previous creations)");
        }
    }
    Util::DecIndent();
}

bool Model::check() {
    getTracer()->trace(Util::TraceLevel::L7_internal, "Checking model consistency");
    Util::IncIndent();
    // before checking the model, creates all necessary internal ModelElements
    _createModelInternalElements();
    bool res = this->_modelChecker->checkAll();
    Util::DecIndent();
    if (res) {
        getTracer()->trace(Util::TraceLevel::L2_results, "End of Model checking: Success");
    } else {
        //std::exception e = new std::exception();
        //getTrace()->traceError() ;
        getTracer()->trace(Util::TraceLevel::L2_results, "End of Model checking: Failed");
    }
    return res;
}

//bool Model::verifySymbol(std::string componentName, std::string expressionName, std::string expression, std::string expressionResult, bool mandatory) {
//    return this->_modelChecker->verifySymbol(componentName, expressionName, expression, expressionResult, mandatory);
//}

Entity* Model::createEntity(std::string name, bool insertIntoModel) {
    // Entity is my FRIEND, therefore Model can access it
    Entity* newEntity = new Entity(this, name, true);
    SimulationEvent *se = _simulation->_createSimulationEvent(); // it's my friend
    se->setEntityCreated(newEntity);
    getOnEvents()->NotifyEntityCreateHandlers(se);
    return newEntity;
}

void Model::removeEntity(Entity* entity) {//, bool collectStatistics) {
    this->_eventManager->NotifyEntityRemoveHandlers(_simulation->_createSimulationEvent()); // it's my friend
    std::string entId = std::to_string(entity->entityNumber());
    this->getElements()->remove(Util::TypeOf<Entity>(), entity);
    getTracer()->trace(/*"Entity " + entId +*/entity->getName() + " was removed from the system");
    entity->~Entity();
}

List<Event*>* Model::getFutureEvents() const {
    return _futureEvents;
}

void Model::setTracer(TraceManager * _traceManager) {
    this->_traceManager = _traceManager;
}

TraceManager * Model::getTracer() const {
    return _traceManager;
}

bool Model::hasChanged() const {
    bool changed = _hasChanged;
    changed &= this->_componentManager->hasChanged();
    changed &= this->_elementManager->hasChanged();
    changed &= this->_modelInfo->hasChanged();
    changed &= this->_modelPersistence->hasChanged();
    return changed;
}
```

```
00342
00343 ComponentManager * Model::getComponents() const {
00344     return _componentManager;
00345 }
00346
00347 List<SimulationControl*>* Model::getControls() const {
00348     return _controls;
00349 }
00350
00351 List<SimulationResponse*>* Model::getResponses() const {
00352     return _responses;
00353 }
00354
00355 OnEventManager * Model::getOnEvents() const {
00356     return _eventManager;
00357 }
00358
00359 ElementManager * Model::getElements() const {
00360     return _elementManager;
00361 }
00362
00363 ModelInfo * Model::getInfos() const {
00364     return _modelInfo;
00365 }
00366
00367 Simulator * Model::getParentSimulator() const {
00368     return _parentSimulator;
00369 }
00370
00371 ModelSimulation * Model::getSimulation() const {
00372     return _simulation;
00373 }
00374
00375 Util::identification Model::getId() const {
00376     return _id;
00377 }
00378
```

8.191 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Model.h File Reference

```
#include <string>
#include "../util/List.h"
#include "ModelComponent.h"
#include "Event.h"
#include "ModelChecker_if.h"
#include "Parser_if.h"
#include "ModelPersistence_if.h"
#include "ElementManager.h"
#include "ComponentManager.h"
#include "TraceManager.h"
#include "OnEventManager.h"
#include "ModelInfo.h"
#include "ModelSimulation.h"
#include "SimulationResponse.h"
#include "SimulationControl.h"
```

Include dependency graph for Model.h:

Classes

- class Model

8.192 Model.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   SimulationModel.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 15:01
00012  */
00013
00014 #ifndef SIMULATIONMODEL_H
00015 #define SIMULATIONMODEL_H
00016
00017 #include <string>
00018
00019 #include "../util/List.h"
00020 #include "ModelComponent.h"
00021 #include "Event.h"
00022 #include "ModelChecker_if.h"
00023 #include "Parser_if.h"
00024 #include "ModelPersistence_if.h"
00025 #include "ElementManager.h"
00026 #include "ComponentManager.h"
00027 #include "TraceManager.h"
00028 #include "OnEventManager.h"
00029 #include "ModelInfo.h"
00030 #include "ModelSimulation.h"
00031 //for PAN
00032 #include "SimulationResponse.h"
00033 #include "SimulationControl.h"
00034
00035 //namespace GenesysKernel {
00036 class Simulator;
00037
00044 class Model {
00045 public:
00046     Model(Simulator* simulator);
00047     virtual ~Model() = default;
00048 public: // model control
00049     //void showReports();
00050     bool save(std::string filename);
00051     bool load(std::string filename);
00052     bool check();
00053     void clear();
00054     void show();
00055     bool insert(ModelElement* elemOrComp);
```

```
    void remove(ModelElement* elemOrComp);
    //bool verifySymbol(std::string componentName, std::string expressionName, std::string expression, std::string expressionResult, bool mandatory); ///< Verifies if a symbol defined in a component (ModelComponent) or element is syntactically valid and addresses existing components or elements. It's used only by and directed by the component that defines the symbol.
    Entity* createEntity(std::string name, bool insertIntoModel);
    void removeEntity(Entity* entity); //, bool collectStatistics);
    void sendEntityToComponent(Entity* entity, Connection* connection, double timeDelay = 0.0);
    void sendEntityToComponent(Entity* entity, ModelComponent* component, double timeDelay = 0.0, unsigned int componentInputNumber = 0);
    double parseExpression(const std::string expression);
    double parseExpression(const std::string expression, bool* success, std::string* errorMessage);
    bool checkExpression(const std::string expression, const std::string expressionName, std::string* errorMessage);
public: // only gets
    Util::identification getId() const;
    bool hasChanged() const;
    // 1:1
    OnEventManager* getOnEvents() const;
    ElementManager* getElements() const;
    ComponentManager* getComponents() const;
    ModelInfo* getInfos() const;
    Simulator* getParentSimulator() const;
    ModelSimulation* getSimulation() const;
    // 1:n
    //List<ModelComponent*>* getComponents() const; ///< Returns the list of components (such as Create, Delay, Dispose, etc.) that make up the simulation model.
    List<Event*>* getFutureEvents() const;
    List<SimulationControl*>* getControls() const;
    List<SimulationResponse*>* getResponses() const;
public: // gets and sets
    void setTracer(TraceManager* _traceManager);
    TraceManager* getTracer() const;
    /*
     * PRIVATE
     */
private:
    void _showConnections() const;
    void _showComponents() const;
    void _showElements() const;
    void _showSimulationControls() const;
    void _showSimulationResponses() const;
    void _createModelInternalElements();
private:
    bool _hasChanged = false;
    bool _isChecked = false; // \todo: Not implemented yet. First, _hasChanged should be trustful
private: // read only public access (gets)
    Util::identification _id;
    Simulator* _parentSimulator;
    // 1:1 (associted classes)
    TraceManager* _traceManager;
    OnEventManager* _eventManager;
    ElementManager* _elementManager;
    ComponentManager* _componentManager;
    ModelInfo* _modelInfo;
    ModelSimulation* _simulation;
    // 1:n
    //List<ModelComponent*>* _components;
    List<Event*>* _futureEvents;
    // for process analyser
    List<SimulationResponse*>* _responses;
    List<SimulationControl*>* _controls;

private: // no public access (no gets / sets)
    ModelChecker_if* _modelChecker;
    ModelPersistence_if* _modelPersistence;
    Parser_if* _parser;
};
//namespace\\}
#endif /* SIMULATIONMODEL_H */
```

8.193 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelChecker_if.h File Reference

```
#include <typeinfo>
#include <string>
```

Include dependency graph for ModelChecker_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelChecker_if

8.194 ModelChecker_if.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ModelChecker_if.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 23 de Agosto de 2018, 15:48
00012  */
00013
00014 #ifndef MODELCHECKER_IF_H
00015 #define MODELCHECKER_IF_H
00016
00017 #include <typeinfo>
00018 #include <string>
00019
00020 //#include "Model.h"
00021 //class Model;
00022
00026 class ModelChecker_if {
00027 public:
00028     virtual bool checkAll() = 0;
00029     virtual bool checkConnected() = 0;
00030     virtual bool checkSymbols() = 0;
00031     virtual bool checkActivationCode() = 0;
00032     virtual bool checkLimits() = 0;
00034     //virtual bool verifySymbol(std::string componentName, std::string expressionName, std::string
       expression, std::string expressionResult, bool mandatory) = 0;
00035 };
00036
00037 #endif /* MODELCHECKER_IF_H */
00038
```

8.195 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelCheckerDefaultImpl1.cpp File Reference

```
#include "ModelCheckerDefaultImpl1.h"
#include "SourceModelComponent.h"
#include "SinkModelComponent.h"
#include "ComponentManager.h"
#include "Simulator.h"
#include <assert.h>
```

Include dependency graph for ModelCheckerDefaultImpl1.cpp:

8.196 ModelCheckerDefaultImpl1.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ModelCheckerDefaultImpl1.cpp
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 8 de Agosto de 2018, 18:44
00012  */
00013
00014 #include "ModelCheckerDefaultImpl1.h"
00015 #include "SourceModelComponent.h"
00016 #include "SinkModelComponent.h"
00017 #include "ComponentManager.h"
00018 #include "Simulator.h"
00019
00020 #include <assert.h>
00021
00022 //using namespace GenesysKernel;
00023
00024 ModelCheckerDefaultImpl1::ModelCheckerDefaultImpl1(Model* model) {
00025     _model = model;
00026 }
00027
00028 bool ModelCheckerDefaultImpl1::checkAll() {
00029     bool res = true;
00030     res &= checkSymbols();
00031     //if (res)
00032     //   res &= checkAndAddInternalLiterals();
00033     //if (res)
00034     //   res &= checkActivationCode();
00035     if (res)
00036         res &= checkLimits();
00037     if (res)
00038         res &= checkConnected();
00039     return res;
00040 }
00041
00042 //bool ModelCheckerDefaultImpl1::checkAndAddInternalLiterals() {
```

```
//    /*  \todo: +-: not implemented yet */
//    return true;
//}

void ModelCheckerDefaultImpl1::_recursiveConnectedTo(PluginManager* pluginManager, ModelComponent* comp, List<ModelComponent*>* visited, List<ModelComponent*>* unconnected, bool* drenoFound) {
    visited->insert(comp);
    _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Connected to component \"" + comp->getName() + "\"");
    Plugin* plugin = pluginManager->find(comp->getClassname());
    assert(plugin != nullptr);
    if (plugin->getPluginInfo()->isSink() || (plugin->getPluginInfo()->isSendTransfer() && comp->getConnections()->size() == 0)) {//(dynamic_cast<SinkModelComponent*> (comp) != nullptr) {
        // it is a sink OR it can send entities throught a transfer and has no nextConnections
        *drenoFound = true;
    } else { // it is not a sink
        if (comp->getConnections()->size() == 0) {
            unconnected->insert(comp);
            _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Component \"" + comp->getName() + "\" is unconnected (not a sink with no next componentes connected to)");
            *drenoFound = false;
        } else {
            ModelComponent* nextComp;
            for (std::list<Connection*>::iterator it = comp->getConnections()->list()->begin(); it != comp->getConnections()->list()->end(); it++) {
                nextComp = (*it)->first;
                if (visited->find(nextComp) == visited->list()->end()) { // not visited yet
                    *drenoFound = false;
                    Util::IncIndent();
                    this->_recursiveConnectedTo(pluginManager, nextComp, visited, unconnected, drenoFound);
                    Util::DecIndent();
                } else {
                    Util::IncIndent();
                    _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Connected to " + nextComp->getName());
                    Util::DecIndent();
                    *drenoFound = true;
                }
            }
        }
    }
}

bool ModelCheckerDefaultImpl1::checkConnected() {
    /*  \todo: +-: not implemented yet */
    _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Checking connected");
    bool resultAll = true;
    PluginManager* pluginManager = this->_model->getParentSimulator()->getPlugins();
    Plugin* plugin;
    Util::IncIndent();
    {
        List<ModelComponent*>* visited = new List<ModelComponent*>();
        List<ModelComponent*>* unconnected = new List<ModelComponent*>();
        ModelComponent* comp;
        for (std::list<ModelComponent*>::iterator it = _model->getComponents()->begin(); it != _model->getComponents()->end(); it++) {
            comp = (*it);
            plugin = pluginManager->find(comp->getClassname());
            assert(plugin != nullptr);
            if (plugin->getPluginInfo()->isSource() || plugin->getPluginInfo()->isReceiveTransfer()) {//(dynamic_cast<SourceModelComponent*> (comp) != nullptr) {
                // it is a source component OR it can receive enetities from transfer
                bool drenoFound = false;
                _recursiveConnectedTo(pluginManager, comp, visited, unconnected, &drenoFound);
            }
        }
        // check if any component remais unconnected
        for (std::list<ModelComponent*>::iterator it = _model->getComponents()->begin(); it != _model->getComponents()->end(); it++) {
            comp = (*it);
            if (visited->find(comp) == visited->list()->end()) { //not found
                resultAll = false;
                _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Component \"" + comp->getName() + "\" is unconnected.");
            }
        }

    }
    Util::DecIndent();
    return resultAll;
}

bool ModelCheckerDefaultImpl1::checkSymbols() {
    bool res = true;
    _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Checking symbols");
    Util::IncIndent();
```

```
    {
        // check components
        _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Components:");
        Util::IncIndent();
        {
            //List<ModelComponent*>* components = _model->getComponents();
            for (std::list<ModelComponent*>::iterator it = _model->getComponents()->begin(); it != _model->getComponents()->end(); it++) {
                res &= (*it)->Check((*it));
            }
        }
        Util::DecIndent();

        // check elements
        if (res) {
            _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Elements:");
            Util::IncIndent();
            {
                std::string elementType;
                bool result;
                ModelElement* element;
                std::string* errorMessage = new std::string();
                std::list<std::string>* elementTypes = _model->getElements()->getElementClassnames();
                for (std::list<std::string>::iterator typeIt = elementTypes->begin(); typeIt != elementTypes->end(); typeIt++) {
                    elementType = (*typeIt);
                    List<ModelElement*>* elements = _model->getElements()->getElementList(elementType);
                    for (std::list<ModelElement*>::iterator it = elements->list()->begin(); it != elements->list()->end(); it++) {
                        element = (*it);
                        // copyed from modelCOmponent. It is not inside the ModelElement::Check because ModelElement has no access to Model to call Tracer
                        _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Checking " + element->getClassname() + ": \"" + element->getName() + "\" (id " + std::to_string(element->getId()) + ")"); //std::to_string(component->_id));
                        Util::IncIndent();
                        {
                            try {
                                result = element->Check((*it), errorMessage);
                                res &= result;
                                if (!result) {
                                    _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Error: Checking has failed with message '" + *errorMessage + "'");
                                }
                            } catch (const std::exception& e) {
                                _model->getTracer()->traceError(e, "Error verifying component " + element->show());
                            }
                        }
                        Util::DecIndent();
                    }
                }
            }
            Util::DecIndent();
        }
    }
    Util::DecIndent();

    return res;
}

bool ModelCheckerDefaultImpl1::checkActivationCode() {
    /*  \todo: +-: not implemented yet */
    _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Checking activation code");
    Util::IncIndent();
    {

    }
    Util::DecIndent();
    return true;
}

bool ModelCheckerDefaultImpl1::checkLimits() {
    bool res = true;
    std::string text;
    unsigned int value, limit;
    LicenceManager *licence = _model->getParentSimulator()->getLicenceManager();
    _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Checking model limits");
    Util::IncIndent();
    {
        value = _model->getComponents()->getNumberOfComponents();
        limit = licence->getModelComponentsLimit();
        res &= value <= limit;
        _model->getTracer()->trace("Model has " + std::to_string(value) + "/" + std::to_string(limit) + " components");
        if (!res) {
```

```
            text = "Model has " + std::to_string(_model->getComponents()->getNumberOfComponents()) + " components, exceding the limit of " + std::to_string(licence->getModelComponentsLimit()) + " components imposed by the current activation code";
            //_model->getTraceManager()->trace(Util::TraceLevel::errors, text);
        } else {
            value = _model->getElements()->getNumberOfElements();
            limit = licence->getModelElementsLimit();
            res &= value <= limit;
            _model->getTracer()->trace("Model has " + std::to_string(value) + "/" + std::to_string(limit) + " elements");
            if (!res) {
                text = "Model has " + std::to_string(_model->getElements()->getNumberOfElements()) + " elements, exceding the limit of " + std::to_string(licence->getModelElementsLimit()) + " elements imposed by the current activation code";
                //_model->getTraceManager()->trace(Util::TraceLevel::errors, text);
            }
        }
        if (!res) {
            _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Error: Checking has failed with message '" + text + "'");
        }
    }
    Util::DecIndent();
    return res;
}
```

8.197 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelChecker↩ DefaultImpl1.h File Reference

```
#include "ModelChecker_if.h"
#include "Model.h"
#include "PluginManager.h"
```

Include dependency graph for ModelCheckerDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelCheckerDefaultImpl1

8.198 ModelCheckerDefaultImpl1.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ModelCheckerDefaultImpl1.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 8 de Agosto de 2018, 18:44
00012  */
00013
00014 #ifndef MODELCHECKERDEFAULTIMPL1_H
00015 #define MODELCHECKERDEFAULTIMPL1_H
00016
00017 #include "ModelChecker_if.h"
00018 #include "Model.h"
00019 #include "PluginManager.h"
00020
00021 //namespace GenesysKernel {
00022
00023 class ModelCheckerDefaultImpl1 : public ModelChecker_if {
00024 public:
00025     ModelCheckerDefaultImpl1(Model* model);
00026     virtual ~ModelCheckerDefaultImpl1() = default;
00027 public:
00028     virtual bool checkAll();
00029     virtual bool checkConnected();
00030     virtual bool checkSymbols();
00031     virtual bool checkActivationCode();
00032     virtual bool checkLimits();
00033     //virtual bool verifySymbol(std::string componentName, std::string expressionName, std::string
      expression, std::string expressionResult, bool mandatory);
00034 private:
00035     void _recursiveConnectedTo(PluginManager* pluginManager, ModelComponent* comp,
      List<ModelComponent*>* visited, List<ModelComponent*>* unconnected, bool* drenoFound);
00036 private:
00037     Model* _model;
00038 };
00039 //namespace\\}
00040 #endif /* MODELCHECKERDEFAULTIMPL1_H */
00041
```

8.199 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelComponent.cpp File Reference

```
#include "ModelComponent.h"
#include "Model.h"
#include "../TraitsKernel.h"
```

Include dependency graph for ModelComponent.cpp:

8.200 ModelComponent.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
/*
 * File:   ModelComponent.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 15:56
 */

#include "ModelComponent.h"
#include "Model.h"
#include "../TraitsKernel.h"

//using namespace GenesysKernel;

ModelComponent::ModelComponent(Model* model, std::string componentTypename, std::string name) : ModelElement(model, componentTypename, name, false) {
    model->getComponents()->insert(this);
}

ModelComponent::~ModelComponent() {
    _parentModel->getComponents()->remove(this);
}

void ModelComponent::Execute(Entity* entity, ModelComponent* component, unsigned int inputNumber) {
    std::string msg = /*"Entity " +std::to_string(entity->entityNumber())*/ entity->getName() + " has arrived at component \"" + component->getName() + "\"";
    if (component->getDescription() != "")
        msg += ": " + component->getDescription();
    // \todo: How can I know the number of inputs?
    if (inputNumber > 0)
        msg += " by input " + std::to_string(inputNumber);
    component->_parentModel->getTracer()->trace(Util::TraceLevel::L6_arrival, msg);
    Util::IncIndent();
    try {
        component->_execute(entity);
    } catch (const std::exception& e) {
        component->_parentModel->getTracer()->traceError(e, "Error executing component " + component->show());
    }
    Util::DecIndent();
}

void ModelComponent::CreateInternalElements(ModelComponent* component) {
    //component->_model->getTraceManager()->trace(Util::TraceLevel::blockArrival, "Writing component \"" + component->_name + "\""); //std::to_string(component->_id));
    try {
        component->_createInternalElements();
    } catch (const std::exception& e) {
        component->_parentModel->getTracer()->traceError(e, "Error creating elements of component " + component->show());
    };
}

std::map<std::string, std::string>* ModelComponent::SaveInstance(ModelComponent* component) {
    component->_parentModel->getTracer()->trace(Util::TraceLevel::L9_mostDetailed, "Writing component \"" + component->getName() + "\""); //std::to_string(component->_id));
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    try {
        fields = component->_saveInstance();
    } catch (const std::exception& e) {
        component->_parentModel->getTracer()->traceError(e, "Error executing component " + component->show());
    }
    return fields;
}

void ModelComponent::setDescription(std::string description) {
    if (_description != description) {
        this->_description = description;
        _hasChanged = true;
    }
}

std::string ModelComponent::getDescription() const {
    return _description;
}

bool ModelComponent::Check(ModelComponent* component) {
    component->_parentModel->getTracer()->trace(Util::TraceLevel::L8_detailed, "Checking " + component->_typename + ": \"" + component->getName() + "\""); //std::to_string(component->_id));
    bool res = false;
```

```
    std::string* errorMessage = new std::string();
    Util::IncIndent();
    {
        try {
            res = component->_check(errorMessage);
            if (!res) {
                component->_parentModel->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Error: Checking has failed with message '" + *errorMessage + "'");
            }
        } catch (const std::exception& e) {
            component->_parentModel->getTracer()->traceError(e, "Error verifying component " + component->show());
        }
    }
    Util::DecIndent();
    return res;
}

ConnectionManager* ModelComponent::getConnections() const {
    return _connections; // \todo How to know if it changes?
}

std::string ModelComponent::show() {
    return ModelElement::show(); // "{id=" + std::to_string(this->_id) + ",name=\""+this->_name + "\"}"; // , nextComponents[]=(" + _nextComponents->show() + ")}";
}

bool ModelComponent::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        _description = LoadField(fields, "caption", DEFAULT.description);
        // Now it should load nextComponents. The problem is that the nextComponent may not be loaded yet.
        // So, what can be done is to temporarily load the ID of the nextComponents, and to wait until all the components have been loaded to update nextComponents based on the temporarilyIDs now being loaded
        //unsigned short nextSize = std::stoi((*fields->find("nextSize")).second);
        //this->_tempLoadNextComponentsIDs = new List<Util::identification>();
        //for (unsigned short i = 0; i < nextSize; i++) {
        //    Util::identification nextId = std::stoi((*fields->find("nextId" + std::to_string(i))).second);
        //    this->_tempLoadNextComponentsIDs->insert(nextId);
        //}
    }
    return res;
}

std::map<std::string, std::string>* ModelComponent::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
    SaveField(fields, "caption", _description, DEFAULT.description);
    if (true) {//(_connections->size() != 1) { // save nextSize only if it is != 1
        SaveField(fields, "nextSize", _connections->size(), DEFAULT.nextSize);
    }
    unsigned short i = 0;
    for (Connection* connection : *_connections->list()) {
        if (_connections->list()->size() == 1) {
            SaveField(fields, "nextId", connection->first->_id, 0);
        } else {
            SaveField(fields, "nextId" + std::to_string(i), connection->first->_id, 0);
        }
        if (connection->second != 0) {//((*it)->second != 0) { // save nextInputNumber only if it is != 0
            SaveField(fields, "nextInputNumber" + std::to_string(i), connection->second, DEFAULT.nextInputNumber);
        }
        i++;
    }
    return fields;
}

void ModelComponent::_createInternalElements() {

}
```

8.201 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelComponent.h File Reference

```
#include <string>
#include <list>
```

```
#include "Plugin.h"
#include "../util/List.h"
#include "Entity.h"
#include "ModelElement.h"
#include "ConnectionManager.h"
```

Include dependency graph for ModelComponent.h:

Classes

- class ModelComponent
- struct ModelComponent::DEFAULT_VALUES

8.202 ModelComponent.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Element.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 15:56
00012  */
00013
00014 #ifndef MODELCOMPONENT_H
00015 #define MODELCOMPONENT_H
00016
00017 #include <string>
00018 #include <list>
00019
00020 #include "Plugin.h"
00021 #include "../util/List.h"
00022 #include "Entity.h"
00023 #include "ModelElement.h"
00024 #include "ConnectionManager.h"
00025 //namespace GenesysKernel {
00026
00027 class Model;
00028
00033 class ModelComponent : public ModelElement {
00034 public:
00035     ModelComponent(Model* model, std::string componentTypename, std::string name = "");
00036     virtual ~ModelComponent();
00037 public:
00038     virtual std::string show();
00039 public:
00040     ConnectionManager* getConnections() const;
00041 public: // static
00042     static void Execute(Entity* entity, ModelComponent* component, unsigned int inputNumber);
00043     //static void InitBetweenReplications(ModelComponent* component);
00044     static void CreateInternalElements(ModelComponent* component);
00045     static bool Check(ModelComponent* component);
00046     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00047     static std::map<std::string, std::string>* SaveInstance(ModelComponent* component);
00048     void setDescription(std::string _description);
00049     std::string getDescription() const;
00050 protected: // pure virtual methods
```

```
    virtual void _execute(Entity* entity) = 0;
protected: // virtual methods
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _createInternalElements();
protected:

    const struct DEFAULT_VALUES {
        unsigned int nextSize = 1;
        unsigned int nextInputNumber = 0;
        std::string description = "";
    } DEFAULT;
    std::string _description = DEFAULT.description;
    ConnectionManager* _connections = new ConnectionManager();
};
//namespace\\}
#endif /* MODELCOMPONENT_H */

```

8.203 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelElement.cpp File Reference

```
#include "ModelElement.h"
#include <iostream>
#include "Model.h"
#include "../TraitsKernel.h"
```

Include dependency graph for ModelElement.cpp:

8.204 ModelElement.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ModelElement.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:40
 */

//#include <typeinfo>
#include "ModelElement.h"
#include <iostream>
#include "Model.h"
#include "../TraitsKernel.h"

//using namespace GenesysKernel;

ModelElement::ModelElement(Model* model, std::string thistypename, std::string name, bool
    insertIntoModel) {
    _id = Util::GenerateNewId(); //GenerateNewIdOfType(thistypename);
    _typename = thistypename;
```

```
    _parentModel = model;
    _reportStatistics = TraitsKernel<ModelElement>::reportStatistics; // \todo: shoould be a parameter before insertIntoModel
    if (name == "")
        _name = thistypename + "_" + std::to_string(Util::GenerateNewIdOfType(thistypename));
    else if (name.substr(name.length() - 1, 1) == "%")
        _name = name.substr(0, name.length() - 1) + std::to_string(Util::GenerateNewIdOfType(thistypename));
    else
        _name = name;
    _hasChanged = false;
    if (insertIntoModel) {
        model->insert(this);
    }
}

bool ModelElement::hasChanged() const {
    return _hasChanged;
}

//ModelElement::ModelElement(const ModelElement &orig) {
//this->_parentModel = orig->_parentModel;
//this->_name = "copy_of_" + orig->_name;
//this->_typename = orig->_typename;
//}

ModelElement::~ModelElement() {
    _parentModel->getTracer()->trace(Util::TraceLevel::L9_mostDetailed, "Removing Element \"" + this->_name + "\" from the model");
    _removeChildrenElements();
    _parentModel->getElements()->remove(this);
}

void ModelElement::_removeChildrenElements() {
    Util::IncIndent();
    {
        for (std::map<std::string, ModelElement*>::iterator it = _childrenElements->begin(); it != _childrenElements->end(); it++) {
            this->_parentModel->getElements()->remove((*it).second);
            (*it).second->~ModelElement();
        }
        _childrenElements->clear();
    }
    Util::DecIndent();

}

void ModelElement::_removeChildElement(std::string key) {
    Util::IncIndent();
    {
        //for (std::map<std::string, ModelElement*>::iterator it = _childrenElements->begin(); it != _childrenElements->end(); it++) {
        std::map<std::string, ModelElement*>::iterator it = _childrenElements->begin();
        while (it != _childrenElements->end()) {
            if ((*it).first == key) {
                this->_parentModel->getElements()->remove((*it).second);
                (*it).second->~ModelElement();
                _childrenElements->erase(it);
                it = _childrenElements->begin();
            }
        }
    }
    Util::DecIndent();
}

bool ModelElement::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = true;
    std::map<std::string, std::string>::iterator it;
    it = fields->find("id");
    it != fields->end() ? this->_id = std::stoi((*it).second) : res = false;
    it = fields->find("name");
    it != fields->end() ? this->_name = (*it).second : this->_name = "";
    //it != fields->end() ? this->_name = (*it).second : res = false;
    it = fields->find("reportStatistics");
    it != fields->end() ? this->_reportStatistics = std::stoi((*it).second) : this->_reportStatistics = TraitsKernel<ModelElement>::reportStatistics;
    return res;
}

std::map<std::string, std::string>* ModelElement::_saveInstance() {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    SaveField(fields, "typename", _typename);
    SaveField(fields, "id", this->_id);
    SaveField(fields, "name", _name);
    SaveField(fields, "reportStatistics", _reportStatistics, TraitsKernel<ModelElement>::reportStatistics);
    return fields;
```

```
}

bool ModelElement::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true; // if there is no ovveride, return true
}

ParserChangesInformation* ModelElement::_getParserChangesInformation() {
    return new ParserChangesInformation(); // if there is no override, return no changes
}

void ModelElement::_initBetweenReplications() {

}

/*
std::list<std::map<std::string,std::string>*>* ModelElement::_saveInstance(std::string type) {
    std::list<std::map<std::string,std::string>*>* fields = ModelElement::_saveInstance();
    fields->push_back(type);
    return fields;
}
 */

std::string ModelElement::show() {
    std::string children = "";
    if (_childrenElements->size() > 0) {
        children = ", children=[";
        for (std::map<std::string, ModelElement*>::iterator it = _childrenElements->begin(); it !=
 _childrenElements->end(); it++) {
            children += (*it).second->getName() + ",";
        }
        children = children.substr(0, children.length() - 1) + "]";
    }
    return "id=" + std::to_string(_id) + ",name=\"" + _name + "\"" + children;
}

std::list<std::string>* ModelElement::getChildrenElementKeys() const {
    std::list<std::string>* result = new std::list<std::string>();
    return result;
}

Util::identification ModelElement::getId() const {
    return _id;
}

void ModelElement::setName(std::string name) {
    // rename every "stuff" related to this element (controls, responses and childrenElements
    if (name != _name) {
        std::string stuffName;
        unsigned int pos;
        for (std::pair<std::string, ModelElement*> child : *_childrenElements) {
            stuffName = child.second->getName();
            pos = stuffName.find(getName(), 0);
            if (pos != std::string::npos) {
                stuffName = stuffName.replace(pos, pos + getName().length(), name);
                child.second->setName(stuffName);
            }
        }

        for (SimulationControl* control : *_parentModel->getControls()->list()) {
            stuffName = control->getName();
            pos = stuffName.find(getName(), 0);
            if (pos != std::string::npos) {
                stuffName = stuffName.replace(pos, pos + getName().length(), name);
                control->setName(stuffName);
            }
        }

        for (SimulationResponse* response : *_parentModel->getResponses()->list()) {
            stuffName = response->getName();
            pos = stuffName.find(getName(), 0);
            if (pos != std::string::npos) {
                stuffName = stuffName.replace(pos, pos + getName().length(), name);
                response->setName(stuffName);
            }
        }
        this->_name = name;
        _hasChanged = true;
    }
}

std::string ModelElement::getName() const {
    return _name;
}

std::string ModelElement::getClassname() const {
    return _typename;
```

```
}

void ModelElement::InitBetweenReplications(ModelElement* element) {
    element->_parentModel->getTracer()->trace("Initing element \"" + element->getName() + "\"", Util::TraceLevel::L8_detailed); //std::to_string(component->_id));
    try {
        element->_initBetweenReplications();
    } catch (const std::exception& e) {
        element->_parentModel->getTracer()->traceError(e, "Error initing component " + element->show());
    };
}

ModelElement* ModelElement::LoadInstance(Model* model, std::map<std::string, std::string>* fields, bool insertIntoModel) {
    std::string name = "";
    if (insertIntoModel) {
        // extracts the name from the fields even before "_laodInstance" and even before construct a new ModelElement in such way when constructing the ModelElement, it's done with the correct name and that correct name is show in trace
        std::map<std::string, std::string>::iterator it = fields->find("name");
        if (it != fields->end())
            name = (*it).second;
    }
    ModelElement* newElement = new ModelElement(model, "ModelElement", name, insertIntoModel);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

std::map<std::string, std::string>* ModelElement::SaveInstance(ModelElement* element) {
    std::map<std::string, std::string>* fields; // = new std::list<std::string>();
    try {
        fields = element->_saveInstance();
    } catch (const std::exception& e) {
        //element->_model->getTrace()->traceError(e, "Error saving anElement " + element->show());
    }
    return fields;
}

bool ModelElement::Check(ModelElement* element, std::string* errorMessage) {
    //    element->_model->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, "Checking " + element->_typename + ": " + element->_name); //std::to_string(element->_id));
    bool res = false;
    Util::IncIndent();
    {
        try {
            res = element->_check(errorMessage);
            if (!res) {
                //                element->_model->getTraceManager()->trace(Util::TraceLevel::errors, "Error: Checking has failed with message '" + *errorMessage + "'");
            }
        } catch (const std::exception& e) {
            //            element->_model->getTraceManager()->traceError(e, "Error verifying element " + element->show());
        }
    }
    Util::DecIndent();
    return res;
}

void ModelElement::CreateInternalElements(ModelElement* element) {
    try {
        Util::IncIndent();
        element->_createInternalElements();
        Util::DecIndent();
    } catch (const std::exception& e) {
        //element->...->_model->getTraceManager()->traceError(e, "Error creating elements of element " + element->show());
    };
}

void ModelElement::_createInternalElements() {

}

void ModelElement::setReportStatistics(bool reportStatistics) {
    if (_reportStatistics != reportStatistics) {
        this->_reportStatistics = reportStatistics;
        _hasChanged = true;
    }
}

bool ModelElement::isReportStatistics() const {
```

```
    return _reportStatistics;
}
```

8.205 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelElement.h File Reference

```
#include <string>
#include <list>
#include <vector>
#include <map>
#include "../util/Util.h"
#include "ParserChangesInformation.h"
#include "PersistentObject_base.h"
```

Include dependency graph for ModelElement.h:

Classes

- class ModelElement

8.206 ModelElement.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ModelElement.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:40
 */

#ifndef MODELELEMENT_H
#define MODELELEMENT_H

#include <string>
#include <list>
#include <vector>
#include <map>
#include "../util/Util.h"

#include "ParserChangesInformation.h"
#include "PersistentObject_base.h"

//namespace GenesysKernel {
class Model;

class ModelElement : PersistentObject_base {
public:
    ModelElement(Model* model, std::string elementTypename, std::string name = "", bool insertIntoModel = true);
    //ModelElement(Model* model, std::string elementTypename, std::string name = "", bool insertIntoModel = true);
    //ModelElement(const ModelElement &orig);
    virtual ~ModelElement();

```

```
public: // get & set
    Util::identification getId() const;
    void setName(std::string name);
    std::string getName() const;
    std::string getClassname() const;
    bool isReportStatistics() const;
    void setReportStatistics(bool reportStatistics);
public: // static
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields, bool insertIntoModel); // \todo: return ModelComponent* ?
    static std::map<std::string, std::string>* SaveInstance(ModelElement* element);
    static bool Check(ModelElement* element, std::string* errorMessage);
    static void CreateInternalElements(ModelElement* element);
    /* This class methood is responsible for invoking the protected method _initBetweenReplication(), which clears all statistics, attributes, counters and other stuff before starting a new repliction */
    static void InitBetweenReplications(ModelElement* element);
public:
    virtual std::string show();
    std::list<std::string>* getChildrenElementKeys() const;
    ModelElement* getChildElement(std::string key) const;
    bool hasChanged() const;
protected:
    void _setChildElement(std::string key, ModelElement* child);
    void _removeChildrenElements();
    void _removeChildElement(std::string key);
protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
protected: // could be overriden by derived classes
    virtual bool _check(std::string* errorMessage);
    virtual ParserChangesInformation* _getParserChangesInformation();
    virtual void _initBetweenReplications();
    virtual void _createInternalElements();
private:
    void _build(Model* model, std::string thistypename, bool insertIntoModel);
private: // name is now private. So changes in name must be throught setName, wich gives oportunity to rename childrenElements, SimulationControls and SimulationResponses
    std::string _name;
protected:
    Util::identification _id;
    std::string _typename;
    bool _reportStatistics;
    bool _hasChanged;
    Model* _parentModel;
protected:
    std::map<std::string, ModelElement*>* _childrenElements = new std::map<std::string, ModelElement*>();
};
//namespace\\}

#endif /* MODELELEMENT_H */

```

8.207 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelInfo.cpp File Reference

```
#include "ModelInfo.h"
#include "Model.h"
```

Include dependency graph for ModelInfo.cpp:

8.208 ModelInfo.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ModelInfo.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 7 de Novembro de 2018, 18:13
 */

#include "ModelInfo.h"
#include "Model.h"

//using namespace GenesysKernel;

ModelInfo::ModelInfo() {
    _name = "Model " + std::to_string(Util::GenerateNewIdOfType<Model>());
}

std::string ModelInfo::show() {
    return "analystName=\"" + this->_analystName + "\"" +
            ",description=\"" + this->_description + "\"" +
            ",name=\"" + this->_name + "\"" +
            ",version=" + this->_version;
}

void ModelInfo::setName(std::string _name) {
    this->_name = _name;
    _hasChanged = true;
}

std::string ModelInfo::getName() const {
    return _name;
}

void ModelInfo::setAnalystName(std::string _analystName) {
    this->_analystName = _analystName;
    _hasChanged = true;
}

std::string ModelInfo::getAnalystName() const {
    return _analystName;
}

void ModelInfo::setDescription(std::string _description) {
    this->_description = _description;
    _hasChanged = true;
}

std::string ModelInfo::getDescription() const {
    return _description;
}
```

```

void ModelInfo::setProjectTitle(std::string _projectTitle) {
    this->_projectTitle = _projectTitle;
    _hasChanged = true;
}

std::string ModelInfo::getProjectTitle() const {
    return _projectTitle;
}

void ModelInfo::setVersion(std::string _version) {
    this->_version = _version;
    _hasChanged = true;
}

std::string ModelInfo::getVersion() const {
    return _version;
}

void ModelInfo::loadInstance(std::map<std::string, std::string>* fields) {
    this->_analystName = LoadField(fields, "analystName");
    this->_description = LoadField(fields, "description");
    this->_name = LoadField(fields, "name");
    this->_projectTitle = LoadField(fields, "projectTitle");
    this->_version = LoadField(fields, "version");
    _hasChanged = false;
}

// \todo:!: implement check method (to check things like terminating condition)

std::map<std::string, std::string>* ModelInfo::saveInstance() {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    SaveField(fields, "typename", "ModelInfo");
    SaveField(fields, "analystName", _analystName);
    SaveField(fields, "description", _description);
    SaveField(fields, "name", this->getName());
    SaveField(fields, "projectTitle", _projectTitle);
    SaveField(fields, "version", _version, "1.0");
    _hasChanged = false;
    return fields;
}

bool ModelInfo::hasChanged() const {
    return _hasChanged;
}
```

8.209 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelInfo.h File Reference

```
#include <string>
#include "../util/Util.h"
```

Include dependency graph for ModelInfo.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelInfo

8.210 ModelInfo.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ModelInfo.h
 * Author: rafael.luiz.cancian
 *
 * Created on 7 de Novembro de 2018, 18:13
 */

#ifndef MODELINFO_H
#define MODELINFO_H

#include <string>
#include "../util/Util.h"

//namespace GenesysKernel {

class ModelInfo {
public:
    ModelInfo();
    virtual ~ModelInfo() = default;
public:
    std::string show();
public: // gets and sets
    void setName(std::string _name);
    std::string getName() const;
    void setAnalystName(std::string _analystName);
    std::string getAnalystName() const;
    void setDescription(std::string _description);
    std::string getDescription() const;
    void setProjectTitle(std::string _projectTitle);
    std::string getProjectTitle() const;
    void setVersion(std::string _version);
    std::string getVersion() const;
public:
    void loadInstance(std::map<std::string, std::string>* fields);
    std::map<std::string, std::string>* saveInstance();
    bool hasChanged() const;
private: // with public access (get & set)
    // model general information
    std::string _name;
    std::string _analystName = "";
    std::string _description = "";
    std::string _projectTitle = "";
    std::string _version = "1.0";
    bool _hasChanged = false;
};
//namespace\\}
#endif /* MODELINFO_H */

```

8.211 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelManager.cpp File Reference

```
#include <fstream>
#include <stdio.h>
#include <iostream>
#include <cstdio>
#include "ModelManager.h"
#include "../util/List.h"
```

```
#include "Simulator.h"
```

Include dependency graph for ModelManager.cpp:

8.212 ModelManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ModelManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 31 de Maio de 2019, 08:37
 */

#include <fstream>
#include <stdio.h>
#include <iostream>
#include <cstdio>

#include "ModelManager.h"
#include "../util/List.h"
#include "Simulator.h"

//using namespace GenesysKernel;

ModelManager::ModelManager(Simulator* simulator) {
    _simulator = simulator;
    _currentModel = nullptr;
}

Model* ModelManager::newModel() {
    _currentModel = new Model(_simulator);
    return _currentModel;
}

void ModelManager::insert(Model* model) {
    _models->insert(model);
    this->_currentModel = model;
    _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Model successfully inserted");
}

void ModelManager::remove(Model* model) {
    _models->remove(model);
    if (_currentModel == model) {
        _currentModel = this->front();
    }
    model->~Model();
    _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Model successfully removed");
}

unsigned int ModelManager::size() {
    return _models->size();
}
```

```
bool ModelManager::saveModel(std::string filename) {
    if (_currentModel != nullptr)
        return _currentModel->save(filename);
    return false;
}

bool ModelManager::loadModel(std::string filename) {
    Model* model = new Model(_simulator);
    bool res = model->load(filename);
    if (res) {
        this->insert(model);
        _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Model successfully loaded");
    } else {
        model->~Model();
        _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Model coud not be loaded");
    }
    return res;
}

bool ModelManager::createFromLanguage(std::string modelSpecification) {
    std::string randomTempFilename = "_tmp";
    const char alphanum[] =
            "0123456789_."
            "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
            "abcdefghijklmnopqrstuvwxyz";
    unsigned short stringLength = sizeof (alphanum) - 1;
    for (unsigned short i = 0; i < 16; i++) {
        randomTempFilename += alphanum[rand() % stringLength];
    }
    std::ofstream savefile;
    savefile.open("tmp_createfromlanguage.tmp" /*randomTempFilename*/, std::ofstream::out);
    savefile « modelSpecification;
    savefile.close();
    bool result = this->loadModel(randomTempFilename);
    //const std::string filename = randomTempFilename;
    std::remove("tmp_createfromlanguage.tmp" /*randomTempFilename*/);
    return result;
}

void ModelManager::setCurrent(Model* model) {
    this->_currentModel = model;
}

Model* ModelManager::current() {
    return _currentModel;
}

Model* ModelManager::front() {
    return _models->front();
}

Model* ModelManager::next() {
    return _models->next();
}
```

8.213 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelManager.h File Reference

```
#include "Model.h"
#include "TraceManager.h"
```

Include dependency graph for ModelManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelManager

8.214 ModelManager.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ModelManager.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 31 de Maio de 2019, 08:37
00012  */
00013
00014 #ifndef MODELMANAGER_H
00015 #define MODELMANAGER_H
00016
00017 #include "Model.h"
00018 #include "TraceManager.h"
00019
00020 //namespace GenesysKernel {
00021
00022 class ModelManager {
00023 public:
00024     ModelManager(Simulator* simulator);
00025     virtual ~ModelManager() = default;
00026 public:
00027     Model* newModel();
00028     void insert(Model* model);
00029     void remove(Model* model);
00030     void setCurrent(Model* model);
00031     bool saveModel(std::string filename);
00032     bool loadModel(std::string filename);
00033     bool createFromLanguage(std::string modelSpecification);
00034     unsigned int size();
00035 public:
```

```
    Model* front();
    Model* current();
    Model* next();
    //Model* end();
private:
    List<Model*>* _models = new List<Model*>();
    Model* _currentModel;
private:
    Simulator* _simulator;
};
//namespace\\}
#endif /* MODELMANAGER_H */

```

8.215 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelPersistence_if.h File Reference

`#include <string>`

Include dependency graph for ModelPersistence_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelPersistence_if

8.216 ModelPersistence_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ModelPersistence_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 24 de Agosto de 2018, 19:22
```

```
 */

#ifndef MODELPERSISTENCE_IF_H
#define MODELPERSISTENCE_IF_H

#include <string>

class ModelPersistence_if {
public:
    // \todo: not a good interface for sure. The Bridge pattern should be a lot better
    virtual bool save(std::string filename) = 0;
    virtual bool load(std::string filename) = 0;
    virtual bool hasChanged() = 0;
private:

};

#endif /* MODELPERSISTENCE_IF_H */

```

8.217 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelPersistence↩ DefaultImpl1.cpp File Reference

```
#include "ModelPersistenceDefaultImpl1.h"
#include <fstream>
#include <ctime>
#include <regex>
#include <cassert>
#include <iterator>
#include "ModelComponent.h"
#include "Simulator.h"
#include "../TraitsKernel.h"
#include "Counter.h"
```

Include dependency graph for ModelPersistenceDefaultImpl1.cpp:

8.218 ModelPersistenceDefaultImpl1.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
```

```
 * File:   ModelPersistenceDefaultImpl1.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 20 de Maio de 2019, 20:41
 */

#include "ModelPersistenceDefaultImpl1.h"
#include <fstream>
#include <ctime>
#include <regex>
#include <cassert>
#include <iterator>

#include "ModelComponent.h"
#include "Simulator.h"
#include "../TraitsKernel.h"
#include "Counter.h"

//using namespace GenesysKernel;

ModelPersistenceDefaultImpl1::ModelPersistenceDefaultImpl1(Model* model) {
    _model = model;
}

std::map<std::string, std::string>* ModelPersistenceDefaultImpl1::_getSimulatorInfoFieldsToSave() {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    SaveField(fields, "typename", "SimulatorInfo");
    SaveField(fields, "name", _model->getParentSimulator()->getName());
    SaveField(fields, "versionNumber", _model->getParentSimulator()->getVersionNumber());
    //SaveField(fields, "version", _model->getParentSimulator()->getVersion());
    return fields;
}

bool ModelPersistenceDefaultImpl1::save(std::string filename) {
    _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Saving file \"" + filename + "\"");
    Util::IncIndent();
    std::list<std::string> *simulatorInfosToSave, *simulationInfosToSave, *modelInfosToSave, *modelElementsToSave, *modelComponentsToSave;
    {
        //bool res = true;
        std::map<std::string, std::string>* fields;
        fields = _getSimulatorInfoFieldsToSave();
        simulatorInfosToSave = _adjustFieldsToSave(fields);
        // save model own infos
        fields = _model->getInfos()->saveInstance();
        modelInfosToSave = _adjustFieldsToSave(fields);
        // save model own infos
        // \todo save modelSimulation fields (breakpoints)
        fields = _model->getSimulation()->saveInstance();
        simulationInfosToSave = _adjustFieldsToSave(fields);
        // save infras
        modelElementsToSave = new std::list<std::string>();
        std::list<std::string>* elementTypenames = _model->getElements()->getElementClassnames();
        const std::string UtilTypeOfCounter = Util::TypeOf<Counter>();
        for (std::list<std::string>::iterator itTypenames = elementTypenames->begin(); itTypenames != elementTypenames->end(); itTypenames++) {
            if ((*itTypenames) != Util::TypeOf<StatisticsCollector>() && (*itTypenames) != UtilTypeOfCounter) { // STATISTICSCOLLECTR and COUNTERs do NOT need to be saved
                List<ModelElement*>* infras = _model->getElements()->getElementList((*itTypenames));
                _model->getTracer()->trace(Util::TraceLevel::L9_mostDetailed, "Writing elements of type \"" + (*itTypenames) + "\":");
                Util::IncIndent();
                {
                    for (std::list<ModelElement*>::iterator it = infras->list()->begin(); it != infras->list()->end(); it++) {
                        _model->getTracer()->trace(Util::TraceLevel::L9_mostDetailed, "Writing " + (*itTypenames) + " \"" + (*it)->getName() + "\"");
                        fields = (*it)->SaveInstance((*it));
                        Util::IncIndent();
                        modelElementsToSave->merge(*_adjustFieldsToSave(fields));
                        Util::DecIndent();
                    }
                }
                Util::DecIndent();
            }
        }
        // save components
        _model->getTracer()->trace(Util::TraceLevel::L9_mostDetailed, "Writing components\":");
        //List<ModelComponent*>* components = this->_model->getComponents();
        modelComponentsToSave = new std::list<std::string>();
        Util::IncIndent();
        {
            for (std::list<ModelComponent*>::iterator it = _model->getComponents()->begin(); it != _model->getComponents()->end(); it++) {
                fields = (*it)->SaveInstance((*it));
                Util::IncIndent();
                modelComponentsToSave->merge(*_adjustFieldsToSave(fields));
```

```
                Util::DecIndent();
            }
        }
        Util::DecIndent();
        // SAVE FILE
        _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Saving file");
        Util::IncIndent();
        {
            // open file
            std::ofstream savefile;
            savefile.open(filename, std::ofstream::out);
            //savefile « "# Genesys simulation model " « std::endl;
            //savefile « "# Last saved on " « dt;
            //savefile « "# simulator infos" « std::endl;
            _saveContent(simulatorInfosToSave, &savefile);
            //savefile « "# model infos" « std::endl;
            _saveContent(modelInfosToSave, &savefile);
            //savefile « "# simulation infos / experimental design" « std::endl;
            _saveContent(simulationInfosToSave, &savefile);
            //savefile « "#" « std::endl;
            //savefile « "# model elements" « std::endl;
            _saveContent(modelElementsToSave, &savefile);
            //savefile « "#" « std::endl;
            //savefile « "# model components" « std::endl;
            _saveContent(modelComponentsToSave, &savefile);
            savefile.close();
        }
        Util::DecIndent();
    }
    Util::DecIndent();
    this->_hasChanged = false;
    return true; // \todo: check if save really saved successfully
}

void ModelPersistenceDefaultImpl1::_saveContent(std::list<std::string>* content, std::ofstream* file) {
    for (std::list<std::string>::iterator it = content->begin(); it != content->end(); it++) {
        *file « (*it) « std::endl; // « std::endl;
    }
}

bool ModelPersistenceDefaultImpl1::_loadFields(std::string line) {
    //std::regex regex{R"([=]+)"}; // split on space R"([\s]+)" \todo: HOW SEPARATOR WITH MORE THAN ONE CHAR
    _model->getTracer()->trace(Util::TraceLevel::L9_mostDetailed, line);
    bool res = true;
    // replaces every "quoted" string by {stringX}
    std::regex regexQuoted("\"([^\"]*)\"");
    auto matches_begin = std::sregex_iterator(line.begin(), line.end(), regexQuoted);
    auto matches_end = std::sregex_iterator();
    std::list<std::string>* strings = new std::list<std::string>();
    for (std::sregex_iterator it = matches_begin; it != matches_end; it++) {
        std::string match_str = (*it).str();
        strings->insert(strings->end(), match_str);
    }
    unsigned short i = 0;
    for (std::list<std::string>::iterator it = strings->begin(); it != strings->end(); it++, i++) {
        std::string match_str = (*it);
        unsigned int pos = line.find(match_str, 0);
        line.replace(pos, match_str.length(), "{string" + std::to_string(i) + "}");
    }
    //
    //i = 0;
    //for (std::list<std::string>::iterator it = strings->begin(); it != strings->end(); it++, i++) {
    //  std::cout « "{string" « std::to_string(i) « "}=" « (*it) « std::endl;
    //}
    std::regex regex{R"([\s]+)"}; // split on " " \todo Should be _lineseparator
    std::sregex_token_iterator tit{line.begin(), line.end(), regex, -1};
    std::list<std::string> lstfields{tit,{}};
    // for each field, separate key and value and form a map
    try {
        std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
        regex = {R"([=]+)"};
        std::vector<std::string> veckeyval; //{it,{}};
        unsigned int i = 0;
        for (std::list<std::string>::iterator it = lstfields.begin(); it != lstfields.end(); it++, i++) {
            //std::cout « (*it) « std::endl;
            tit = {(*it).begin(), (*it).end(), regex, -1};
            veckeyval = {tit,{}};
            veckeyval[0] = trim((veckeyval[0]));
            if (veckeyval[0] != "") { // it should always be, rigth? \todo case for assert?
                if (veckeyval.size() > 1) {
                    veckeyval[1] = trim((veckeyval[1]));
                    if (veckeyval[1].substr(0, 1) == "\"" && veckeyval[1].substr(veckeyval[1].length() - 1, 1) == "\"") { // remove ""
                        veckeyval[1] = veckeyval[1].substr(1, veckeyval[1].length() - 2);
```

```
                    }
                    veckeyval[1] = this->_convertLineseparatorReplacementBacktoLineseparator(veckeyval[1]);
                    fields->emplace(veckeyval[0], veckeyval[1]);
                } else {
                    if (i == 0) {
                        fields->emplace("id", veckeyval[0]);
                    } else if (i == 1) {
                        fields->emplace("typename", veckeyval[0]);
                    } else if (i == 2) {
                        //  veckeyval[0] = veckeyval[0].substr(1, veckeyval[0].length() - 2);
                        //}
                        //veckeyval[0] = this->_convertLineseparatorReplacementBacktoLineseparator(veckeyval[0]);
                        fields->emplace("name", veckeyval[0]);
                    } else {
                        fields->emplace(veckeyval[0], "");
                    }
                }
            }
        }
        // replaces back {stringX} by the strings themselves
        i = 0;
        std::map<std::string, std::string>::iterator fieldsIt = fields->begin();
        for (std::list<std::string>::iterator it = strings->begin(); it != strings->end(); it++, i++) {
            while ((*fieldsIt).second != "{string" + std::to_string(i) + "}") {
                fieldsIt++;
            }
            (*fieldsIt).second = (*it).substr(1, (*it).length() - 2);
            fieldsIt = fields->begin();
        }

        // now the map<str,str> is ready. Look for the right class to load it
        Util::IncIndent();
        {
            std::string thistypename = (*fields->find("typename")).second;
            _model->getTracer()->trace(Util::TraceLevel::L7_internal, "loading " + thistypename + "");
            if (thistypename == "SimulatorInfo") {
                this->_loadSimulatorInfoFields(fields);
            } else if (thistypename == "ModelInfo") {
                _model->getInfos()->loadInstance(fields);
            } else if (thistypename == "ModelSimulation") {
                _model->getSimulation()->loadInstance(fields);
            } else {
                // this should be a ModelComponent or ModelElement.
                ModelElement* newUselessElement = ModelElement::LoadInstance(_model, fields, false);
                if (newUselessElement != nullptr) {
                    // \todo how free newUselessElement without invoking destructor?
                    Plugin* plugin = this->_model->getParentSimulator()->getPlugins()->find(thistypename);
                    if (plugin != nullptr) {
                        res = plugin->loadAndInsertNew(_model, fields);
                        // save fields for components, in order to allow to connect components after all of them have been loaded
                        if (res && plugin->getPluginInfo()->isComponent()) {
                            _componentFields->insert(_componentFields->end(), fields);
                        }
                    } else {
                        _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Error loading file: Could not identity typename \"" + thistypename + "\"");
                        res = false;
                    }
                } else {
                    _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Error loading file: Could not identity typename \"" + thistypename + "\"");
                    res = false;
                }
            }
        }
        Util::DecIndent();
    } catch (...) {

    }
    return res;
}

void ModelPersistenceDefaultImpl1::_loadSimulatorInfoFields(std::map<std::string, std::string>* fields) {
    unsigned int savedVersionNumber = LoadField(fields, "versionNumber", 0);
    unsigned int simulatorVersionNumber = _model->getParentSimulator()->getVersionNumber();
    if (savedVersionNumber != simulatorVersionNumber) {
        _model->getTracer()->trace("WARNING: The version of the saved model differs from the simulator. Loading may not be possible", Util::TraceLevel::L3_errorRecover);
    }
}

```

```
bool ModelPersistenceDefaultImpl1::load(std::string filename) {
    bool res = true;
    _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Loading file \"" + filename + "\"");
    Util::IncIndent();
    _componentFields->clear();
    {
        std::ifstream modelFile;
        std::string inputLine;
        try {
            modelFile.open(filename);
            while (getline(modelFile, inputLine) && res) {
                inputLine = trim(inputLine);
                if (inputLine.substr(0, 1) != "#" && !inputLine.empty()) {
                    //Util::IncIndent();
                    res &= _loadFields(inputLine);
                    //Util::DecIndent();
                }
            }
            modelFile.close();
        } catch (...) {
            _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Error loading file \"" + filename + "\"");
        }
    }
    // check if something was loaded
    res &= (_model->getComponents()->getNumberOfComponents() > 0) & (_model->getElements()->getNumberOfElements() > 0);
    if (res) {
        //
        // CONNECT LOADED COMPONENTS (must wait for all components to be loaded so they can be connected)
        //
        ComponentManager* cm = _model->getComponents();
        _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Connecting loaded components");
        Util::IncIndent();
        {
            for (std::list<std::map<std::string, std::string>*>::iterator it = _componentFields->begin(); it != _componentFields->end(); it++) {
                std::map<std::string, std::string>* fields = (*it);
                // find the component
                ModelComponent* thisComponent = nullptr;
                Util::identification thisId = std::stoi((*fields->find("id")).second);
                for (std::list<ModelComponent*>::iterator itcomp = cm->begin(); itcomp != cm->end(); itcomp++) {
                    if ((*itcomp)->getId() == thisId) {
                        thisComponent = (*itcomp);
                        break; // end inner for loop
                    }
                }
                assert(thisComponent != nullptr);

                // find the next components connected with this one
                unsigned short nextSize = 1;
                if (fields->find("nextSize") != fields->end()) { // found nextSize
                    nextSize = std::stoi((*fields->find("nextSize")).second);
                }
                for (unsigned short i = 0; i < nextSize; i++) {
                    Util::identification nextId;
                    if (nextSize == 1) {
                        if (fields->find("nextId") != fields->end()) {
                            nextId = std::stoi((*fields->find("nextId")).second);
                        } else {
                            nextId = std::stoi((*fields->find("nextId" + std::to_string(i))).second);
                        }
                    } else {
                        nextId = std::stoi((*fields->find("nextId" + std::to_string(i))).second);
                    }
                    unsigned short nextInputNumber = 0; // default value if it is not found bellow
                    if (fields->find("nextInputNumber" + std::to_string(i)) != fields->end())
                        nextInputNumber = std::stoi((*fields->find("nextInputNumber" + std::to_string(i))).second);
                    ModelComponent* nextComponent = nullptr;
                    for (std::list<ModelComponent*>::iterator itcomp = cm->begin(); itcomp != cm->end(); itcomp++) {// connect the components
                        if ((*itcomp)->getId() == nextId) { // connect the components
                            nextComponent = (*itcomp);
                            thisComponent->getConnections()->insert(nextComponent, nextInputNumber);
                            _model->getTracer()->trace(Util::TraceLevel::L8_detailed, thisComponent->getName() + "<" + std::to_string(i) + ">" + " --> " + nextComponent->getName() + "<" + std::to_string(nextInputNumber) + ">");
                            break;
                        }
                    }
                    assert(nextComponent != nullptr);
                }

            }
```

```
        }
        Util::DecIndent();
        _model->getTracer()->trace(Util::TraceLevel::L7_internal, "File successfully loaded with " + std::to_string(_model->getComponents()->getNumberOfComponents()) + " components and " + std::to_string(_model->getElements()->getNumberOfElements()) + " elements");
    }
    Util::DecIndent();
    if (res) {
        _hasChanged = false;
    }
    return res;
}

std::string ModelPersistenceDefaultImpl1::_convertLineseparatorReplacementBacktoLineseparator(std::string str) {
    size_t index = str.find(_fieldseparatorReplacement, 0);
    while (index != std::string::npos) {
        str.replace(index, _fieldseparatorReplacement.length(), _fieldseparator);
        /* Advance index forward so the next iteration doesn't pick it up as well. */
        index = str.find(_fieldseparatorReplacement, index + _fieldseparator.length());
    }
    return str;
}

std::string ModelPersistenceDefaultImpl1::_convertLineseparatorToLineseparatorReplacement(std::string str) {
    size_t index = str.find(_fieldseparator, 0);
    while (index != std::string::npos) {
        str.replace(index, _fieldseparator.length(), _fieldseparatorReplacement);
        /* Advance index forward so the next iteration doesn't pick it up as well. */
        index = str.find(_fieldseparator, index + _fieldseparatorReplacement.length());
    }
    return str;
}

std::list<std::string>* ModelPersistenceDefaultImpl1::_adjustFieldsToSave(std::map<std::string, std::string>* fields) {
    std::list<std::string>* newList = new std::list<std::string>();
    std::string newStr, strV2003, idV2003, typenameV2003, nameV2003, nextIDV2004; //, strV210329;
    idV2003 = "0";
    std::string attrValue;
    for (std::map<std::string, std::string>::iterator it = fields->begin(); it != fields->end(); it++) {
        //newStr += (*it).first + "=" + (*it).second + this->_linefieldseparator;
        if ((*it).first == "id")
            idV2003 = (*it).second;
        else if ((*it).first == "typename") {
            if ((*it).second.substr(0, 1) == "\"" && (*it).second.substr((*it).second.length() - 1, 1) == "\"")
                typenameV2003 = (*it).second.substr(1, (*it).second.length() - 2); // remove "" on the bounders
            else
                typenameV2003 = (*it).second;
        } else if ((*it).first == "name")
            nameV2003 = (*it).second; //_convertLineseparatorToLineseparatorReplacement((*it).second);
        else if ((*it).first.find("nextId", 0) != std::string::npos)
            nextIDV2004 += (*it).first + "=" + (*it).second + _fieldseparator;
        else if ((*it).first.find("nextInputNumber", 0) != std::string::npos)
            nextIDV2004 += (*it).first + "=" + (*it).second + _fieldseparator;
        else if ((*it).first.find("nextSize", 0) != std::string::npos)
            nextIDV2004 = (*it).first + "=" + (*it).second + _fieldseparator + nextIDV2004;
        else {
            // version V210329: (*it).second should NEVER contain _linefieldseparator. So, replace it by _linefieldseparatorReplacement
            attrValue = (*it).second; //_convertLineseparatorToLineseparatorReplacement((*it).second);
            strV2003 += (*it).first + "=" + attrValue + _fieldseparator;
        }
    }
    while (idV2003.length() < 3)
        idV2003 += _fieldseparator;
    while (typenameV2003.length() < 10)
        typenameV2003 += _fieldseparator;
    strV2003 = idV2003 + _fieldseparator + typenameV2003 + _fieldseparator + nameV2003 + _fieldseparator + strV2003 + nextIDV2004;
    _model->getTracer()->trace(Util::TraceLevel::L8_detailed, strV2003); //newStr
    newList->push_back(strV2003); //newStr
    return newList;
}

bool ModelPersistenceDefaultImpl1::hasChanged() {
    return _hasChanged;
}
```

8.219 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelPersistence↩ DefaultImpl1.h File Reference

```
#include "ModelPersistence_if.h"
#include "Model.h"
```

Include dependency graph for ModelPersistenceDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelPersistenceDefaultImpl1

8.220 ModelPersistenceDefaultImpl1.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ModelPersistenceDefaultImpl1.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 20 de Maio de 2019, 20:41
00012  */
00013
00014 #ifndef MODELPERSISTENCEDEFAULTIMPL1_H
00015 #define MODELPERSISTENCEDEFAULTIMPL1_H
00016
00017 #include "ModelPersistence_if.h"
00018 #include "Model.h"
00019
00020 //namespace GenesysKernel {
00021
00022 class ModelPersistenceDefaultImpl1 : public ModelPersistence_if {
00023 public:
00024     ModelPersistenceDefaultImpl1(Model* model);
00025     virtual ~ModelPersistenceDefaultImpl1() = default;
00026 public:
00027     virtual bool save(std::string filename);
```

```
    virtual bool load(std::string filename);
    virtual bool hasChanged();
private:
    void _saveContent(std::list<std::string>* content, std::ofstream* file);
    bool _loadFields(std::string line);
    void _loadSimulatorInfoFields(std::map<std::string, std::string>* fields);
    std::list<std::string>* _adjustFieldsToSave(std::map<std::string, std::string>* fields);
    std::string _convertLineseparatorToLineseparatorReplacement(std::string str);
    std::string _convertLineseparatorReplacementBacktoLineseparator(std::string str);
    std::map<std::string, std::string>* _getSimulatorInfoFieldsToSave();
private:
    std::list<std::map<std::string, std::string>*>* _componentFields = new
 std::list<std::map<std::string, std::string>*>();
private:
    Model* _model = nullptr;
    bool _hasChanged = false;
    std::string _fieldseparator = " "; // fields are separated by space
    std::string _fieldseparatorReplacement = "\\_"; // in cases where spaces are in data to be saved,
 they are replaced by this pattern, so there will not be separators inside data (replacement is \_)
};
//namespace\\}
#endif /* MODELPERSISTENCEDEFAULTIMPL1_H */

```

8.221 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelSimulation.cpp File Reference

```
#include "ModelSimulation.h"
#include <iostream>
#include <cassert>
#include <chrono>
#include "Model.h"
#include "Simulator.h"
#include "SourceModelComponent.h"
#include "StatisticsCollector.h"
#include "Counter.h"
#include "SimulationControl.h"
#include "ComponentManager.h"
#include "../TraitsKernel.h"
```

Include dependency graph for ModelSimulation.cpp:

8.222 ModelSimulation.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */;
```

```
/*
 * File:   ModelSimulation.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 7 de Novembro de 2018, 18:04
 */

#include "ModelSimulation.h"
#include <iostream>
#include <cassert>
#include <chrono>
#include "Model.h"
#include "Simulator.h"
#include "SourceModelComponent.h"
#include "StatisticsCollector.h"
#include "Counter.h"
#include "SimulationControl.h"
#include "ComponentManager.h"
#include "../TraitsKernel.h"

//using namespace GenesysKernel;

ModelSimulation::ModelSimulation(Model* model) {
    _model = model;
    _info = model->getInfos();
    _statsCountersSimulation->setSortFunc([](const ModelElement* a, const ModelElement * b) {
        return a->getId() < b->getId();
    });
    _simulationReporter = new TraitsKernel<SimulationReporter_if>::Implementation(this, model, this->_statsCountersSimulation);
}

std::string ModelSimulation::show() {
    return "numberOfReplications=" + std::to_string(_numberOfReplications) +
            ",replicationLength=" + std::to_string(_replicationLength) + " " + Util::StrTimeUnitLong(this->_replicationLengthTimeUnit) +
            ",terminatingCondition=\"" + this->_terminatingCondition + "\"" +
            ",warmupTime=" + std::to_string(this->_warmUpPeriod) + " " + Util::StrTimeUnitLong(this->_warmUpPeriodTimeUnit);
}

bool ModelSimulation::_isReplicationEndCondition() {
    bool finish = _model->getFutureEvents()->size() == 0;
    finish |= _model->parseExpression(_terminatingCondition) != 0.0;
    if (_model->getFutureEvents()->size() > 0 && !finish) {
        // replication length has not been achieve (sor far), but next event will happen after that, so it's just fine to set tnow as the replicationLength
        finish |= _model->getFutureEvents()->front()->getTime() > _replicationLength * _replicationTimeScaleFactorToBase;
    }
    return finish;
}

void ModelSimulation::_traceReplicationEnded() {
    std::string causeTerminated = "";
    if (_model->getFutureEvents()->empty()) {
        causeTerminated = "event queue is empty";
    } else if (_stopRequested) {
        causeTerminated = "user requested to stop";
    } else if (_model->getFutureEvents()->front()->getTime() > _replicationLength) {
        causeTerminated = "replication length " + std::to_string(_replicationLength) + " " + Util::StrTimeUnitLong(_replicationLengthTimeUnit) + " was achieved";
    } else if (_model->parseExpression(_terminatingCondition)) {
        causeTerminated = "termination condition was achieved";
    } else causeTerminated = "unknown";
    std::chrono::duration<double> duration = std::chrono::system_clock::now() - this->_startRealSimulationTimeReplication;
    std::string message = "Replication " + std::to_string(_currentReplicationNumber) + " of " + std::to_string(_numberOfReplications) + " has finished with last event at time " + std::to_string(_simulatedTime) + " " + Util::StrTimeUnitLong(_replicationBaseTimeUnit) + " because " + causeTerminated + "; Elapsed time " + std::to_string(duration.count()) + " seconds.";
    _model->getTracer()->trace(Util::TraceLevel::L2_results, message);
}

SimulationEvent* ModelSimulation::_createSimulationEvent(void* thiscustomObject) {
    SimulationEvent* se = new SimulationEvent();
    se->currentComponent = _currentComponent;
    se->currentEntity = _currentEntity;
    se->currentEvent = _currentEvent;
    se->currentInputNumber = _currentInputNumber;
    se->currentReplicationNumber = _currentReplicationNumber;
    se->customObject = thiscustomObject;
    se->_isPaused = this->_isPaused;
    se->_isRunning = this->_isRunning;
    se->pauseRequested = _pauseRequested;
    se->simulatedTime = _simulatedTime;
```

```
    se->stopRequested = _stopRequested;
    return se;
}

void ModelSimulation::start() {
    if (!_simulationIsInitiated) { // begin of a new simulation
        Util::SetIndent(0); //force indentation
        if (!_model->check()) {
            _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, "Model check failed. Cannot start simulation.");
            return;
        }
        _initSimulation();
    }
    if (_isPaused) { // continue after a pause
        _model->getTracer()->trace("Replication resumed", Util::TraceLevel::L3_errorRecover);
        _model->getOnEvents()->NotifySimulationResumeHandlers(_createSimulationEvent());
    }
    _isRunning = true;
    _isPaused = false;
    bool replicationEnded;
    do {
        if (!_replicationIsInitiaded) {
            Util::SetIndent(1);
            _model->getOnEvents()->NotifyReplicationStartHandlers(_createSimulationEvent());
            _initReplication();
            Util::IncIndent();
        }
        do { // this is the main simulation loop
            _stepSimulation();
            replicationEnded = _isReplicationEndCondition();
        } while (!replicationEnded && !_pauseRequested);
        if (replicationEnded) {
            Util::SetIndent(1); // force
            _replicationEnded();
            _currentReplicationNumber++;
            if (_currentReplicationNumber <= _numberOfReplications) {
                if (_pauseOnReplication) {
                    _model->getTracer()->trace("End of replication. Simulation is paused.", Util::TraceLevel::L7_internal);
                    _pauseRequested = true;
                }
            } else {
                _pauseRequested = false;
            }
        }
    } while (_currentReplicationNumber <= _numberOfReplications && !_pauseRequested);
    // all replications done (or paused during execution)
    if (!_pauseRequested) { // done
        _simulationEnded();
    } else { // paused
        _model->getTracer()->trace("Replication paused", Util::TraceLevel::L3_errorRecover);
        _model->getOnEvents()->NotifySimulationPausedHandlers(_createSimulationEvent());
        _pauseRequested = false;
        _isPaused = true;
    }
    _isRunning = false;
    _pauseRequested = false;
}

void ModelSimulation::_simulationEnded() {
    _simulationIsInitiated = false;
    if (this->_showReportsAfterSimulation)
        _simulationReporter->showSimulationStatistics(); //_cStatsSimulation);
    Util::DecIndent();
    // clear current event
    _currentEntity = nullptr;
    _currentComponent = nullptr;
    //
    std::chrono::duration<double> duration = std::chrono::system_clock::now() - this->_startRealSimulationTimeSimulation;
    _model->getTracer()->trace(Util::TraceLevel::L2_results, "Simulation of model \"" + _info->getName() + "\" has finished. Elapsed time " + std::to_string(duration.count()) + " seconds.");
    _model->getOnEvents()->NotifySimulationEndHandlers(_createSimulationEvent());
}

void ModelSimulation::_replicationEnded() {
    _traceReplicationEnded();
    _model->getOnEvents()->NotifyReplicationEndHandlers(_createSimulationEvent());
    if (this->_showReportsAfterReplication)
        _simulationReporter->showReplicationStatistics();
    //_simulationReporter->showSimulationResponses();
    _actualizeSimulationStatistics();
    _replicationIsInitiaded = false;
}

```

```
00168 void ModelSimulation::_actualizeSimulationStatistics() {
00169     //\todo: should not be only CSTAT and COUNTER, but any element that generateReportInformation
00170     const std::string UtilTypeOfStatisticsCollector = Util::TypeOf<StatisticsCollector>();
00171     const std::string UtilTypeOfCounter = Util::TypeOf<Counter>();
00172
00173     StatisticsCollector *sc, *scSim;
00174     List<ModelElement*>* cstats =
      _model->getElements()->getElementList(Util::TypeOf<StatisticsCollector>());
00175     for (std::list<ModelElement*>::iterator itMod = cstats->list()->begin(); itMod !=
      cstats->list()->end(); itMod++) {
00176         sc = dynamic_cast<StatisticsCollector*> ((*itMod));
00177         scSim = nullptr;
00178         for (std::list<ModelElement*>::iterator itSim = _statsCountersSimulation->list()->begin();
      itSim != _statsCountersSimulation->list()->end(); itSim++) {
00179             if ((*itSim)->getClassname() == UtilTypeOfStatisticsCollector) {
00180                 if ((*itSim)->getName() == _cte_stCountSimulNamePrefix + sc->getName() &&
      dynamic_cast<StatisticsCollector*> (*itSim)->getParent() == sc->getParent()) { // found
00181                     scSim = dynamic_cast<StatisticsCollector*> (*itSim);
00182                     break;
00183                 }
00184             }
00185         }
00186         //scSim = dynamic_cast<StatisticsCollector*> (*(this->_statsCountersSimulation->find((*it))));
00187         if (scSim == nullptr) {
00188             // this is a cstat created during the last replication
00189             //      scSim = new StatisticsCollector(_model->elements(), sc->name(), sc->getParent());
00190             _statsCountersSimulation->insert(scSim);
00191         }
00192         assert(scSim != nullptr);
00193         scSim->getStatistics()->getCollector()->addValue(sc->getStatistics()->average());
00194     }
00195     Counter *cnt; //, *cntSim;
00196     List<ModelElement*>* counters = _model->getElements()->getElementList(Util::TypeOf<Counter>());
00197     for (std::list<ModelElement*>::iterator itMod = counters->list()->begin(); itMod !=
      counters->list()->end(); itMod++) {
00198         cnt = dynamic_cast<Counter*> ((*itMod));
00199         //cntSim = nullptr;
00200         scSim = nullptr;
00201         for (std::list<ModelElement*>::iterator itSim = _statsCountersSimulation->list()->begin();
      itSim != _statsCountersSimulation->list()->end(); itSim++) {
00202             if ((*itSim)->getClassname() == UtilTypeOfStatisticsCollector) {
00203                 //_model->getTraceManager()->trace(Util::TraceLevel::simulation, (*itSim)->getName() +
      " == "+_cte_stCountSimulNamePrefix + cnt->getName());
00204                 if ((*itSim)->getName() == _cte_stCountSimulNamePrefix + cnt->getName() &&
      dynamic_cast<StatisticsCollector*> (*itSim)->getParent() == cnt->getParent()) {
00205                     // found
00206                     scSim = dynamic_cast<StatisticsCollector*> (*itSim);
00207                     break;
00208                 }
00209             }
00210             /*
00211             if ((*itSim)->getTypename() == UtilTypeOfCounter) {
00212             if ((*itSim)->getName() == _cte_stCountSimulNamePrefix + cnt->getName() &&
      dynamic_cast<Counter*> (*itSim)->getParent() == cnt->getParent()) {
00213                 // found
00214                 cntSim = dynamic_cast<Counter*> (*itSim);
00215                 break;
00216             }
00217             }
00218              */
00219         }
00220         /*
00221         assert(cntSim != nullptr);
00222         cntSim->incCountValue(cnt->getCountValue());
00223          */
00224         assert(scSim != nullptr);
00225         scSim->getStatistics()->getCollector()->addValue(cnt->getCountValue());
00226     }
00227     //_model->getTracer()->trace(Util::TraceLevel::L2_results, "Simulation stats: " +
      std::to_string(this->_statsCountersSimulation->size()));
00228 }
00229
00230 void ModelSimulation::_showSimulationHeader() {
00231     TraceManager* tm = _model->getTracer();
00232     tm->traceReport("\n-----------------------------------------------------");
00233     // simulator infos
00234     tm->traceReport(_model->getParentSimulator()->getName());
00235     tm->traceReport(_model->getParentSimulator()->getLicenceManager()->showLicence());
00236     tm->traceReport(_model->getParentSimulator()->getLicenceManager()->showLimits());
00237     // model infos
00238     tm->traceReport("Analyst Name: " + _info->getAnalystName());
00239     tm->traceReport("Project Title: " + _info->getProjectTitle());
00240     tm->traceReport("Number of Replications: " + std::to_string(_numberOfReplications));
00241     tm->traceReport("Replication Length: " + std::to_string(_replicationLength) + " " +
      Util::StrTimeUnitLong(_replicationLengthTimeUnit));
00242     tm->traceReport("Replication/Report Base TimeUnit: " +
      Util::StrTimeUnitLong(_replicationBaseTimeUnit));
```

```
    //tm->traceReport(Util::TraceLevel::simulation, "");
    // model controls and responses
    std::string controls;
    for (SimulationControl* control : * _model->getControls()->list()) {
        controls += control->getName() + "(" + control->getType() + ")=" + std::to_string(control->getValue()) + ", ";
    }
    controls = controls.substr(0, controls.length() - 2);
    tm->traceReport("> Simulation controls: " + controls);
    std::string responses;
    for (std::list<SimulationResponse*>::iterator it = _model->getResponses()->list()->begin(); it != _model->getResponses()->list()->end(); it++) {
        responses += (*it)->getName() + "(" + (*it)->getType() + "), ";
    }
    responses = responses.substr(0, responses.length() - 2);
    tm->traceReport("> Simulation responses: " + responses);
    tm->traceReport("");
}

void ModelSimulation::_initSimulation() {
    _startRealSimulationTimeSimulation = std::chrono::system_clock::now();
    _showSimulationHeader();
    _model->getTracer()->trace(Util::TraceLevel::L5_event, "");
    _model->getTracer()->trace(Util::TraceLevel::L5_event, "Simulation of model \"" + _info->getName() + "\" is starting.");
    // defines the time scale factor to adjust replicatonLength to replicationBaseTime
    _replicationTimeScaleFactorToBase = Util::TimeUnitConvert(this->_replicationLengthTimeUnit, this->_replicationBaseTimeUnit);
    // copy all CStats and Counters (used in a replication) to CStats and counters for the whole simulation
    // \todo: Should not be CStats and Counters, but any element that generates report importation
    this->_statsCountersSimulation->clear();
    StatisticsCollector* cstat;
    List<ModelElement*>* cstats = _model->getElements()->getElementList(Util::TypeOf<StatisticsCollector>());
    for (std::list<ModelElement*>::iterator it = cstats->list()->begin(); it != cstats->list()->end(); it++) {
        cstat = dynamic_cast<StatisticsCollector*> ((*it));
        // this new CSat should NOT be inserted into the model
        StatisticsCollector* newCStatSimul = new StatisticsCollector(_model, _cte_stCountSimulNamePrefix + cstat->getName(), cstat->getParent(), false);
        //_model->elements()->remove(Util::TypeOf<StatisticsCollector>(), newCStatSimul); // remove from model, since it is automatically inserted by the constructor
        this->_statsCountersSimulation->insert(newCStatSimul);
    }
    // copy all Counters (used in a replication) to Counters for the whole simulation
    // \todo: Counters in replication should be converted into CStats in simulation. Each value counted in a replication should be added in a CStat for Stats.
    Counter* counter;
    List<ModelElement*>* counters = _model->getElements()->getElementList(Util::TypeOf<Counter>());
    for (std::list<ModelElement*>::iterator it = counters->list()->begin(); it != counters->list()->end(); it++) {
        counter = dynamic_cast<Counter*> ((*it));
        // adding a counter
        /*
        Counter* newCountSimul = new Counter(_cte_stCountSimulNamePrefix + counter->getName(), counter->getParent());
        this->_statsCountersSimulation->insert(newCountSimul);
         */
        // addin a cstat (to stat the counts)
        StatisticsCollector* newCStatSimul = new StatisticsCollector(_model, _cte_stCountSimulNamePrefix + counter->getName(), counter->getParent(), false);
        this->_statsCountersSimulation->insert(newCStatSimul);
    }
    _simulationIsInitiated = true; // \todo Check the uses of _simulationIsInitiated and when it should be set to false
    _replicationIsInitiaded = false;
    _currentReplicationNumber = 1;
    _model->getOnEvents()->NotifySimulationStartHandlers(_createSimulationEvent());
}

void ModelSimulation::_initReplication() {
    _startRealSimulationTimeReplication = std::chrono::system_clock::now();
    TraceManager* tm = _model->getTracer();
    tm->trace(Util::TraceLevel::L5_event, "");
    tm->trace(Util::TraceLevel::L2_results, "Replication " + std::to_string(_currentReplicationNumber) + " of " + std::to_string(_numberOfReplications) + " is starting.");
    _model->getFutureEvents()->clear();
    _model->getElements()->getElementList("Entity")->clear();
    _simulatedTime = 0.0;
    // init all components between replications
    Util::IncIndent();
    tm->trace(Util::TraceLevel::L8_detailed, "Initing Replication");
    for (std::list<ModelComponent*>::iterator it = _model->getComponents()->begin(); it != _model->getComponents()->end(); it++) {
        ModelComponent::InitBetweenReplications((*it));
    }
```

```
    // init all elements between replications
    std::list<std::string>* elementTypes = _model->getElements()->getElementClassnames();
    for (std::string elementType : *elementTypes) {//std::list<std::string>::iterator typeIt = elementTypes->begin(); typeIt != elementTypes->end(); typeIt++) {
        List<ModelElement*>* elements = _model->getElements()->getElementList(elementType);
        for (ModelElement* element : *elements->list()) {//std::list<ModelElement*>::iterator it = elements->list()->begin(); it != elements->list()->end(); it++) {
            ModelElement::InitBetweenReplications(element);
        }
    }
    Util::DecIndent();
    //}
    Util::ResetIdOfType(Util::TypeOf<Entity>());
    Util::ResetIdOfType(Util::TypeOf<Event>());
    // insert first creation events
    SourceModelComponent *source;
    Entity *newEntity;
    Event *newEvent;
    double creationTime;
    unsigned int numToCreate;
    for (std::list<ModelComponent*>::iterator it = _model->getComponents()->begin(); it != _model->getComponents()->end(); it++) {
        source = dynamic_cast<SourceModelComponent*> (*it);
        if (source != nullptr) {
            creationTime = source->getFirstCreation();
            numToCreate = source->getEntitiesPerCreation();
            for (unsigned int i = 1; i <= numToCreate; i++) {
                newEntity = _model->createEntity(source->getEntityType()->getName() + "_%", false);
                newEntity->setEntityType(source->getEntityType());
                newEvent = new Event(creationTime, newEntity, (*it));
                _model->getFutureEvents()->insert(newEvent);
            }
            source->setEntitiesCreated(numToCreate);
        }
    }
    if (this->_initializeStatisticsBetweenReplications) {
        _initStatistics();
    }
    this->_replicationIsInitiaded = true; // \todo Check the uses of _replicationIsInitiaded and when it should be set to false
}

void ModelSimulation::_initStatistics() {
    StatisticsCollector* cstat;
    List<ModelElement*>* list = _model->getElements()->getElementList(Util::TypeOf<StatisticsCollector>());
    for (std::list<ModelElement*>::iterator it = list->list()->begin(); it != list->list()->end(); it++) {
        cstat = (StatisticsCollector*) (*it);
        cstat->getStatistics()->getCollector()->clear();
    }
    Counter* counter;
    list = _model->getElements()->getElementList(Util::TypeOf<Counter>());
    for (std::list<ModelElement*>::iterator it = list->list()->begin(); it != list->list()->end(); it++) {
        counter = (Counter*) (*it);
        counter->clear();
    }
}

void ModelSimulation::_checkWarmUpTime(Event* nextEvent) {
    double warmupTime = Util::TimeUnitConvert(_warmUpPeriodTimeUnit, _replicationBaseTimeUnit);
    warmupTime *= _warmUpPeriod;
    if (warmupTime > 0.0 && _model->getSimulation()->getSimulatedTime() <= warmupTime && nextEvent->getTime() > warmupTime) {// warmuTime. Time to initStats
        _model->getTracer()->trace(Util::TraceLevel::L7_internal, "Warmup time reached. Statistics are being reseted.");
        _initStatistics();
    }
}

void ModelSimulation::_stepSimulation() {
    // process one single event
    Event* nextEvent;
    nextEvent = _model->getFutureEvents()->front();
    if (_warmUpPeriod > 0.0)
        _checkWarmUpTime(nextEvent);
    if (nextEvent->getTime() <= _replicationLength * _replicationTimeScaleFactorToBase) {
        if (_checkBreakpointAt(nextEvent)) {
            this->_pauseRequested = true;
        } else {
            _model->getFutureEvents()->pop_front();
            _model->getOnEvents()->NotifyReplicationStepHandlers(_createSimulationEvent());
            _processEvent(nextEvent);
        }
    } else {
        this->_simulatedTime = _replicationLength;
```

```
    }
}

bool ModelSimulation::_checkBreakpointAt(Event* event) {
    bool res = false;
    SimulationEvent* se = _createSimulationEvent();
    if (_breakpointsOnComponent->find(event->getComponent()) != _breakpointsOnComponent->list()->end()) {
        if (_justTriggeredBreakpointsOnComponent == event->getComponent()) {
            _justTriggeredBreakpointsOnComponent = nullptr;
        } else {
            _justTriggeredBreakpointsOnComponent = event->getComponent();
            _model->getOnEvents()->NotifyBreakpointHandlers(se);
            _model->getTracer()->trace("Breakpoint found at component '" + event->getComponent()->getName() + "'. Replication is paused.", Util::TraceLevel::L5_event);

            res = true;
        }
    }
    if (_breakpointsOnEntity->find(event->getEntity()) != _breakpointsOnEntity->list()->end()) {
        if (_justTriggeredBreakpointsOnEntity == event->getEntity()) {
            _justTriggeredBreakpointsOnEntity = nullptr;
        } else {
            _justTriggeredBreakpointsOnEntity = event->getEntity();
            _model->getOnEvents()->NotifyBreakpointHandlers(se);
            _model->getTracer()->trace("Breakpoint found at entity '" + event->getEntity()->getName() + "'. Replication is paused.", Util::TraceLevel::L5_event);
            res = true;
        }
    }
    double time;
    for (std::list<double>::iterator it = _breakpointsOnTime->list()->begin(); it != _breakpointsOnTime->list()->end(); it++) {
        time = (*it);
        if (_simulatedTime < time && event->getTime() >= time) {
            if (_justTriggeredBreakpointsOnTime == time) { // just trrigered this breakpoint
                _justTriggeredBreakpointsOnTime = 0.0;
            } else {
                _justTriggeredBreakpointsOnTime = time;
                _model->getOnEvents()->NotifyBreakpointHandlers(se);
                _model->getTracer()->trace("Breakpoint found at time '" + std::to_string(event->getTime()) + "'. Replication is paused.", Util::TraceLevel::L5_event);
                return true;
            }
        }
    }
    return res;
}

void ModelSimulation::_processEvent(Event* event) {
    //  _model->getTracer()->traceSimulation(Util::TraceLevel::75_event, event->time(), event->entity(), event->component(), "");
    _model->getTracer()->trace(Util::TraceLevel::L5_event, "Event {" + event->show() + "}");
    Util::IncIndent();
    _model->getTracer()->trace(Util::TraceLevel::L8_detailed, "Entity " + event->getEntity()->show());
    this->_currentEvent = event;
    // next three lines could be removed since currentEvet is now available
    this->_currentEntity = event->getEntity();
    this->_currentComponent = event->getComponent();
    this->_currentInputNumber = event->getComponentInputNumber();
    //assert(_simulatedTime <= event->getTime()); // _simulatedTime only goes forward (futureEvents is chronologically sorted
    if (event->getTime() >= _simulatedTime) { // the philosophical approach taken is: if the next event is in the past, lets just assume it's happening rigth now...
        _simulatedTime = event->getTime();
    }
    _model->getOnEvents()->NotifyProcessEventHandlers(_createSimulationEvent());
    try {
        //event->getComponent()->Execute(event->getEntity(), event->getComponent()); // Execute is static
        ModelComponent::Execute(event->getEntity(), event->getComponent(), event->getComponentInputNumber());
    } catch (std::exception *e) {
        _model->getTracer()->traceError(*e, "Error on processing event (" + event->show() + ")");
    }
    if (_pauseOnEvent) {
        _pauseRequested = true;
    }
    Util::DecIndent();
}

void ModelSimulation::pause() {
    _pauseRequested = true;
}

void ModelSimulation::step() {
    bool savedPauseRequest = _pauseRequested;
```

```
    _pauseRequested = true;
    this->start();
    _pauseRequested = savedPauseRequest;
}

void ModelSimulation::stop() {
    this->_isPaused = false;
    this->_isRunning = false;
    this->_replicationIsInitiaded = false;
    this->_simulationIsInitiated = false;
}

void ModelSimulation::setPauseOnEvent(bool _pauseOnEvent) {
    this->_pauseOnEvent = _pauseOnEvent;
}

bool ModelSimulation::isPauseOnEvent() const {
    return _pauseOnEvent;
}

void ModelSimulation::setInitializeStatistics(bool _initializeStatistics) {
    this->_initializeStatisticsBetweenReplications = _initializeStatistics;
}

bool ModelSimulation::isInitializeStatistics() const {
    return _initializeStatisticsBetweenReplications;
}

void ModelSimulation::setInitializeSystem(bool _initializeSystem) {
    this->_initializeSystem = _initializeSystem;
}

bool ModelSimulation::isInitializeSystem() const {
    return _initializeSystem;
}

void ModelSimulation::setStepByStep(bool _stepByStep) {
    this->_stepByStep = _stepByStep;
}

bool ModelSimulation::isStepByStep() const {
    return _stepByStep;
}

void ModelSimulation::setPauseOnReplication(bool _pauseOnReplication) {
    this->_pauseOnReplication = _pauseOnReplication;
}

bool ModelSimulation::isPauseOnReplication() const {
    return _pauseOnReplication;
}

double ModelSimulation::getSimulatedTime() const {
    return _simulatedTime;
}

bool ModelSimulation::isRunning() const {
    return _isRunning;
}

unsigned int ModelSimulation::getCurrentReplicationNumber() const {
    return _currentReplicationNumber;
}

ModelComponent* ModelSimulation::getCurrentComponent() const {
    return _currentComponent;
}

Entity* ModelSimulation::getCurrentEntity() const {
    return _currentEntity;
}

void ModelSimulation::setReporter(SimulationReporter_if* _simulationReporter) {
    this->_simulationReporter = _simulationReporter;
}

SimulationReporter_if* ModelSimulation::getReporter() const {
    return _simulationReporter;
}

unsigned int ModelSimulation::getCurrentInputNumber() const {
    return _currentInputNumber;
}

void ModelSimulation::setShowReportsAfterReplication(bool showReportsAfterReplication) {
    this->_showReportsAfterReplication = showReportsAfterReplication;
}
```

```

bool ModelSimulation::isShowReportsAfterReplication() const {
    return _showReportsAfterReplication;
}

void ModelSimulation::setShowReportsAfterSimulation(bool showReportsAfterSimulation) {
    this->_showReportsAfterSimulation = showReportsAfterSimulation;
}

bool ModelSimulation::isShowReportsAfterSimulation() const {
    return _showReportsAfterSimulation;
}

List<double>* ModelSimulation::getBreakpointsOnTime() const {
    return _breakpointsOnTime;
}

List<Entity*>* ModelSimulation::getBreakpointsOnEntity() const {
    return _breakpointsOnEntity;
}

List<ModelComponent*>* ModelSimulation::getBreakpointsOnComponent() const {
    return _breakpointsOnComponent;
}

bool ModelSimulation::isPaused() const {
    return _isPaused;
}

void ModelSimulation::setNumberOfReplications(unsigned int _numberOfReplications) {
    this->_numberOfReplications = _numberOfReplications;
    _hasChanged = true;
}

unsigned int ModelSimulation::getNumberOfReplications() const {
    return _numberOfReplications;
}

void ModelSimulation::setReplicationLength(double _replicationLength) {
    this->_replicationLength = _replicationLength;
    _hasChanged = true;
}

double ModelSimulation::getReplicationLength() const {
    return _replicationLength;
}

void ModelSimulation::setReplicationLengthTimeUnit(Util::TimeUnit _replicationLengthTimeUnit) {
    this->_replicationLengthTimeUnit = _replicationLengthTimeUnit;
    _hasChanged = true;
}

Util::TimeUnit ModelSimulation::getReplicationLengthTimeUnit() const {
    return _replicationLengthTimeUnit;
}

void ModelSimulation::setWarmUpPeriod(double _warmUpPeriod) {
    this->_warmUpPeriod = _warmUpPeriod;
    _hasChanged = true;
}

double ModelSimulation::getWarmUpPeriod() const {
    return _warmUpPeriod;
}

void ModelSimulation::setWarmUpPeriodTimeUnit(Util::TimeUnit _warmUpPeriodTimeUnit) {
    this->_warmUpPeriodTimeUnit = _warmUpPeriodTimeUnit;
    _hasChanged = true;
}

Util::TimeUnit ModelSimulation::getWarmUpPeriodTimeUnit() const {
    return _warmUpPeriodTimeUnit;
}

void ModelSimulation::setTerminatingCondition(std::string _terminatingCondition) {
    this->_terminatingCondition = _terminatingCondition;
    _hasChanged = true;
}

std::string ModelSimulation::getTerminatingCondition() const {
    return _terminatingCondition;
}

void ModelSimulation::loadInstance(std::map<std::string, std::string>* fields) {
    this->_numberOfReplications = LoadField(fields, "numberOfReplications",
    DEFAULT.numberOfReplications);
    this->_replicationLength = LoadField(fields, "replicationLength", DEFAULT.replicationLength);
```

```
    this->_replicationLengthTimeUnit = LoadField(fields, "replicationLengthTimeUnit", DEFAULT.replicationLengthTimeUnit);
    this->_replicationBaseTimeUnit = LoadField(fields, "replicationBaseTimeUnit", DEFAULT.replicationBeseTimeUnit);
    this->_terminatingCondition = LoadField(fields, "terminatingCondition", DEFAULT.terminatingCondition);
    this->_warmUpPeriod = LoadField(fields, "warmUpTime", DEFAULT.warmUpPeriod);
    this->_warmUpPeriodTimeUnit = LoadField(fields, "warmUpTimeTimeUnit", DEFAULT.warmUpPeriodTimeUnit);
    _hasChanged = false;
}

// \todo:!: implement check method (to check things like terminating condition)

std::map<std::string, std::string>* ModelSimulation::saveInstance() {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    SaveField(fields, "typename", "ModelSimulation");
    SaveField(fields, "name", "fixed experiment name in ModelSimulation"); //getName());
    SaveField(fields, "numberOfReplications", _numberOfReplications, DEFAULT.numberOfReplications);
    SaveField(fields, "replicationLength", _replicationLength, DEFAULT.replicationLength);
    SaveField(fields, "replicationLengthTimeUnit", _replicationLengthTimeUnit, DEFAULT.replicationLengthTimeUnit);
    SaveField(fields, "replicationBaseTimeUnit", _replicationBaseTimeUnit, DEFAULT.replicationBeseTimeUnit);
    SaveField(fields, "terminatingCondition", _terminatingCondition, DEFAULT.terminatingCondition);
    SaveField(fields, "warmUpTime", _warmUpPeriod, DEFAULT.warmUpPeriod);
    SaveField(fields, "warmUpTimeTimeUnit", _warmUpPeriodTimeUnit, DEFAULT.warmUpPeriodTimeUnit);
    _hasChanged = false;
    return fields;
}

Event* ModelSimulation::getCurrentEvent() const {
    return _currentEvent;
}

void ModelSimulation::setShowSimulationResposesInReport(bool _showSimulationResposesInReport) {
    this->_showSimulationResposesInReport = _showSimulationResposesInReport;
}

bool ModelSimulation::isShowSimulationResposesInReport() const {
    return _showSimulationResposesInReport;
}

void ModelSimulation::setShowSimulationControlsInReport(bool _showSimulationControlsInReport) {
    this->_showSimulationControlsInReport = _showSimulationControlsInReport;
}

bool ModelSimulation::isShowSimulationControlsInReport() const {
    return _showSimulationControlsInReport;
}

void ModelSimulation::setReplicationReportBaseTimeUnit(Util::TimeUnit _replicationReportBaseTimeUnit) {
    this->_replicationBaseTimeUnit = _replicationReportBaseTimeUnit;
}

Util::TimeUnit ModelSimulation::getReplicationBaseTimeUnit() const {
    return _replicationBaseTimeUnit;
}
```

8.223 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ModelSimulation.h File Reference

```
#include <chrono>
#include "Event.h"
#include "Entity.h"
#include "ModelInfo.h"
#include "SimulationReporter_if.h"
#include "OnEventManager.h"
```

Include dependency graph for ModelSimulation.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ModelSimulation

8.224 ModelSimulation.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ModelSimulation.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 7 de Novembro de 2018, 18:04
00012  */
00013
00014 #ifndef MODELSIMULATION_H
00015 #define MODELSIMULATION_H
00016
00017 #include <chrono>
00018 #include "Event.h"
00019 #include "Entity.h"
00020 #include "ModelInfo.h"
00021 #include "SimulationReporter_if.h"
00022 #include "OnEventManager.h"
00023 //#include "Counter.h"
00024 //namespace GenesysKernel {
00025
00026 //#include "Model.h" // for friend functions
00027 class Model;
00028
00033 class ModelSimulation { // 202104 to be subjected to SimulationScenario
00034 public:
00035     ModelSimulation(Model* model);
00036     virtual ~ModelSimulation() = default;
00037 public:
00038     std::string show();
00039 public: // simulation control
00040     void start();
00041     void pause();
00042     void step();
```

```
    void stop();
public: // old modelInfos
    void setNumberOfReplications(unsigned int _numberOfReplications);
    unsigned int getNumberOfReplications() const;
    void setReplicationLength(double _replicationLength);
    double getReplicationLength() const;
    void setReplicationLengthTimeUnit(Util::TimeUnit _replicationLengthTimeUnit);
    Util::TimeUnit getReplicationLengthTimeUnit() const;
    void setReplicationReportBaseTimeUnit(Util::TimeUnit _replicationReportBaseTimeUnit);
    Util::TimeUnit getReplicationBaseTimeUnit() const;
    void setWarmUpPeriod(double _warmUpPeriod);
    double getWarmUpPeriod() const;
    void setWarmUpPeriodTimeUnit(Util::TimeUnit _warmUpPeriodTimeUnit);
    Util::TimeUnit getWarmUpPeriodTimeUnit() const;
    void setTerminatingCondition(std::string _terminatingCondition);
    std::string getTerminatingCondition() const;
public: // gets and sets
    void setPauseOnEvent(bool _pauseOnEvent);
    bool isPauseOnEvent() const;
    void setStepByStep(bool _stepByStep);
    bool isStepByStep() const;
    void setInitializeStatistics(bool _initializeStatistics);
    bool isInitializeStatistics() const;
    void setInitializeSystem(bool _initializeSystem);
    bool isInitializeSystem() const;
    void setPauseOnReplication(bool _pauseBetweenReplications);
    bool isPauseOnReplication() const;
public: // gets and sets ModelSubParts
    void setReporter(SimulationReporter_if* _simulationReporter);
    SimulationReporter_if* getReporter() const;
public: // only gets
    double getSimulatedTime() const;
    bool isRunning() const;
    bool isPaused() const;
    unsigned int getCurrentReplicationNumber() const;
    ModelComponent* getCurrentComponent() const;
    Entity* getCurrentEntity() const;
    unsigned int getCurrentInputNumber() const;
    void setShowReportsAfterReplication(bool showReportsAfterReplication);
    bool isShowReportsAfterReplication() const;
    void setShowReportsAfterSimulation(bool showReportsAfterSimulation);
    bool isShowReportsAfterSimulation() const;
    List<double>* getBreakpointsOnTime() const;
    List<Entity*>* getBreakpointsOnEntity() const;
    List<ModelComponent*>* getBreakpointsOnComponent() const;
public:
    void loadInstance(std::map<std::string, std::string>* fields);
    std::map<std::string, std::string>* saveInstance();
    Event* getCurrentEvent() const;
    void setShowSimulationResposesInReport(bool _showSimulationResposesInReport);
    bool isShowSimulationResposesInReport() const;
    void setShowSimulationControlsInReport(bool _showSimulationControlsInReport);
    bool isShowSimulationControlsInReport() const;
    /*
     * PRIVATE
     */
private: // simulation control
    void _initSimulation();
    void _initReplication();
    void _initStatistics();
    void _checkWarmUpTime(Event* nextEvent);
    void _stepSimulation();
    void _replicationEnded();
    void _simulationEnded();
    void _processEvent(Event* event);
private:
    bool _checkBreakpointAt(Event* event);
    bool _isReplicationEndCondition();
    void _actualizeSimulationStatistics();
    void _showSimulationHeader();
    void _traceReplicationEnded();
private:
    SimulationEvent* _createSimulationEvent(void* thiscustomObject = nullptr);
    //friend Entity* Model::createEntity(std::string name, bool insertIntoModel); //\TODO: make it
work (only friend functions, not the entire class)
    //friend void Model::removeEntity(Entity* entity);
    //friend void Model::sendEntityToComponent(Entity* entity, ModelComponent* component, double
timeDelay, unsigned int componentInputNumber);
    friend class Model;
private:
    double _simulatedTime = 0.0;
    // \todo: list of double double _breakOnTimes;
    // \todo: list of modules _breakOnModules;
    bool _stepByStep = false;
    bool _pauseOnReplication = false;
    bool _pauseOnEvent = false;
    bool _initializeStatisticsBetweenReplications = true;
```

```
    bool _initializeSystem = true;
    bool _isRunning = false;
    bool _isPaused = false;
    bool _pauseRequested = false;
    bool _stopRequested = false;
    bool _simulationIsInitiated = false;
    bool _replicationIsInitiaded = false;
    bool _showReportsAfterSimulation = true;
    bool _showReportsAfterReplication = true;
    bool _showSimulationControlsInReport = false;
    bool _showSimulationResposesInReport = false;
private:

    const struct DEFAULT_VALUES {
        unsigned int numberOfReplications = 1;
        double replicationLength = 3600.0; // by default, 3600 s = 1.0 h
        Util::TimeUnit replicationLengthTimeUnit = Util::TimeUnit::second;
        Util::TimeUnit replicationBeseTimeUnit = Util::TimeUnit::second;
        Util::TimeUnit replicationReportBaseTimeUnit = Util::TimeUnit::second;
        double warmUpPeriod = replicationLength * 0.10; // 0.0;
        Util::TimeUnit warmUpPeriodTimeUnit = Util::TimeUnit::second;
        std::string terminatingCondition = "";
        bool showReportsAfterSimulation = true;
        bool initializeStatisticsBetweenReplications = true;
        bool initializeSystem = true;
        bool showReportsAfterReplication = true;
        bool showSimulationControlsInReport = false;
        bool showSimulationResposesInReport = false;
    } DEFAULT;
    unsigned int _numberOfReplications = DEFAULT.numberOfReplications;
    double _replicationLength = DEFAULT.replicationLength;
    Util::TimeUnit _replicationLengthTimeUnit = DEFAULT.replicationLengthTimeUnit;
    Util::TimeUnit _replicationBaseTimeUnit = DEFAULT.replicationReportBaseTimeUnit;
    double _warmUpPeriod = DEFAULT.warmUpPeriod;
    Util::TimeUnit _warmUpPeriodTimeUnit = DEFAULT.warmUpPeriodTimeUnit;
    std::string _terminatingCondition = DEFAULT.terminatingCondition;
    bool _hasChanged = false;
    double _replicationTimeScaleFactorToBase; // a scale that converts ReplicationLenghtTimeUnit to
 ReplicationBaseTimeUnit. Future events are in "times" of unit ReplicationBaseTimeUnit
    std::chrono::system_clock::time_point _startRealSimulationTimeSimulation;
    std::chrono::system_clock::time_point _startRealSimulationTimeReplication;
private:
    // currenEntity, currentComponent and currentInputNumber could be taken throught currentEvent only
    Entity* _currentEntity = nullptr;
    ModelComponent* _currentComponent = nullptr;
    unsigned int _currentInputNumber = 0;
    Event* _currentEvent = nullptr;
    unsigned int _currentReplicationNumber;
private:
    const std::string _cte_stCountSimulNamePrefix = ""; //Simul.";
private:
    Model* _model;
    ModelInfo* _info;
    SimulationReporter_if* _simulationReporter;
    List<ModelElement*>* _statsCountersSimulation = new List<ModelElement*>();
    List<double>* _breakpointsOnTime = new List<double>();
    List<ModelComponent*>* _breakpointsOnComponent = new List<ModelComponent*>();
    List<Entity*>* _breakpointsOnEntity = new List<Entity*>();
    double _justTriggeredBreakpointsOnTime = 0.0;
    ModelComponent* _justTriggeredBreakpointsOnComponent = nullptr;
    Entity* _justTriggeredBreakpointsOnEntity = nullptr;
};
//namespace\\}
#endif /* MODELSIMULATION_H */
```

8.225 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/OnEventManager.cpp File Reference

```
#include "OnEventManager.h"
```

Include dependency graph for OnEventManager.cpp:

8.226 OnEventManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   OnEventManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 7 de Novembro de 2018, 12:28
 */

#include "OnEventManager.h"

//using namespace GenesysKernel;

OnEventManager::OnEventManager() {
}

void OnEventManager::_addOnHandler(List<simulationEventHandler>* list, simulationEventHandler EventHandler) {
    if (list->find(EventHandler) == list->list()->end())
        list->insert(EventHandler);
}

void OnEventManager::addOnReplicationStartHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onReplicationStartHandlers, EventHandler);
}

void OnEventManager::addOnReplicationStepHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onReplicationStepHandlers, EventHandler);
}

void OnEventManager::addOnProcessEventHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onEntityMoveHandlers, EventHandler);
}

void OnEventManager::addOnEntityCreateHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onEntityCreateHandlers, EventHandler);
}

void OnEventManager::addOnEntityMoveHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onEntityMoveHandlers, EventHandler);
}

```

```
void OnEventManager::addOnEntityRemoveHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onEntityRemoveHandlers, EventHandler);
}

void OnEventManager::addOnReplicationEndHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onReplicationEndHandlers, EventHandler);
}

void OnEventManager::addOnSimulationStartHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onSimulationStartHandlers, EventHandler);
}

void OnEventManager::addOnSimulationPausedHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onSimulationPausedHandlers, EventHandler);
}

void OnEventManager::addOnSimulationResumeHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onSimulationResumeHandlers, EventHandler);
}

void OnEventManager::addOnSimulationEndHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onSimulationEndHandlers, EventHandler);
}

void OnEventManager::addOnBreakpointHandler(simulationEventHandler EventHandler) {
    _addOnHandler(_onBreakpointHandlers, EventHandler);
}

void OnEventManager::_NotifyHandlers(List<simulationEventHandler>* list, SimulationEvent* se) {
    for (std::list<simulationEventHandler>::iterator it = list->list()->begin(); it !=
 list->list()->end(); it++) {
        (*it)(se);
    }
}

void OnEventManager::_NotifyHandlerMethods(List<simulationEventHandlerMethod>* list, SimulationEvent*
 se) {
    for (std::list<simulationEventHandlerMethod>::iterator it = list->list()->begin(); it !=
 list->list()->end(); it++) {
        (*it)(se);
    }
}

void OnEventManager::NotifyReplicationStartHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onReplicationStartHandlers, se);
    this->_NotifyHandlerMethods(this->_onReplicationStartHandlerMethods, se);
}

void OnEventManager::NotifyReplicationStepHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onReplicationStepHandlers, se);
    this->_NotifyHandlerMethods(this->_onReplicationStepHandlerMethods, se);
}

void OnEventManager::NotifyReplicationEndHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onReplicationEndHandlers, se);
    this->_NotifyHandlerMethods(this->_onReplicationEndHandlerMethods, se);
}

void OnEventManager::NotifyEntityCreateHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onEntityCreateHandlers, se);
    this->_NotifyHandlerMethods(this->_onEntityCreateHandlerMethods, se);
}

void OnEventManager::NotifyEntityMoveHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onEntityMoveHandlers, se);
    this->_NotifyHandlerMethods(this->_onEntityMoveHandlerMethods, se);
}

void OnEventManager::NotifyEntityRemoveHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onEntityRemoveHandlers, se);
    this->_NotifyHandlerMethods(this->_onEntityRemoveHandlerMethods, se);
}

void OnEventManager::NotifyProcessEventHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onProcessEventHandlers, se);
    this->_NotifyHandlerMethods(this->_onProcessEventHandlerMethods, se);
}

void OnEventManager::NotifySimulationStartHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onSimulationStartHandlers, se);
    this->_NotifyHandlerMethods(this->_onSimulationStartHandlerMethods, se);
}

void OnEventManager::NotifySimulationPausedHandlers(SimulationEvent* se) {
    this->_NotifyHandlers(this->_onSimulationPausedHandlers, se);
    this->_NotifyHandlerMethods(this->_onSimulationPausedHandlerMethods, se);
}
```

```
00130
00131 void OnEventManager::NotifySimulationResumeHandlers(SimulationEvent* se) {
00132     this->_NotifyHandlers(this->_onSimulationResumeHandlers, se);
00133     this->_NotifyHandlerMethods(this->_onSimulationResumeHandlerMethods, se);
00134 }
00135
00136 void OnEventManager::NotifySimulationEndHandlers(SimulationEvent* se) {
00137     this->_NotifyHandlers(this->_onSimulationEndHandlers, se);
00138     this->_NotifyHandlerMethods(this->_onSimulationEndHandlerMethods, se);
00139 }
00140
00141 void OnEventManager::NotifyBreakpointHandlers(SimulationEvent* se) {
00142     this->_NotifyHandlers(this->_onBreakpointHandlers, se);
00143     this->_NotifyHandlerMethods(this->_onBreakpointHandlerMethods, se);
00144 }
```

8.227 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/OnEventManager.h File Reference

```
#include "../util/List.h"
#include "Event.h"
```

Include dependency graph for OnEventManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationEvent
- class OnEventManager

Typedefs

- typedef void(∗ simulationEventHandler) (SimulationEvent ∗)
- typedef std::function< void(SimulationEvent ∗) > simulationEventHandlerMethod

8.227.1 Typedef Documentation

8.227.1.1 simulationEventHandler `typedef void(* simulationEventHandler) (SimulationEvent *)`

Definition at line 132 of file OnEventManager.h.

8.227.1.2 simulationEventHandlerMethod `typedef std::function<void(SimulationEvent*) > simulationEventHandlerM`

Definition at line 134 of file OnEventManager.h.

8.228 OnEventManager.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   OnEventManager.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 7 de Novembro de 2018, 12:28
00012  */
00013
00014 #ifndef ONEVENTMANAGER_H
00015 #define ONEVENTMANAGER_H
00016
00017 #include "../util/List.h"
00018 #include "Event.h"
00019
00020 //namespace GenesysKernel {
00021 /*
00022  * used to get and set values no matter the class (for process analyser)
00023  * should be a wait to invoke a getter or setter no matter the class (a pointer to a member function
       without specifying the class
00024  */
00025 //typedef double (*memberFunctionGetDoubleVarHandler)(); //template<> ... typedef double
       (T::*getDoubleVarHandler)() or something like that
00026 //typedef void (*memberFunctionSetDoubleVarHandler)(double);
00027
00030 class SimulationEvent {
00031 public:
00032     void setSimulatedTime(double simulatedTime) {
00033         this->simulatedTime = simulatedTime;
00034     }
00035     double getSimulatedTime() const {
00036         return simulatedTime;
00037     }
00038     void setStopRequested(bool stopRequested) {
00039         this->stopRequested = stopRequested;
00040     }
00041     bool isStopRequested() const {
00042         return stopRequested;
00043     }
00044     void setPauseRequested(bool pauseRequested) {
00045         this->pauseRequested = pauseRequested;
00046     }
00047     bool isPauseRequested() const {
00048         return pauseRequested;
00049     }
00050     void setPaused(bool Paused) {
00051         _isPaused = Paused;
00052     }
00053     bool isPaused() const {
00054         return _isPaused;
00055     }
00056     void setRunning(bool Running) {
00057         _isRunning = Running;
00058     }
```

```
    bool isRunning() const {
        return _isRunning;
    }
    void setCustomObject(void* customObject) {
        this->customObject = customObject;
    }
    void* getCustomObject() const {
        return customObject;
    }
    void setCurrentReplicationNumber(unsigned int currentReplicationNumber) {
        this->currentReplicationNumber = currentReplicationNumber;
    }
    unsigned int getCurrentReplicationNumber() const {
        return currentReplicationNumber;
    }
    void setCurrentEvent(Event* currentEvent) {
        this->currentEvent = currentEvent;
    }
    Event* getCurrentEvent() const {
        return currentEvent;
    }
    void setCurrentInputNumber(unsigned int currentInputNumber) {
        this->currentInputNumber = currentInputNumber;
    }
    unsigned int getCurrentInputNumber() const {
        return currentInputNumber;
    }
    void setCurrentComponent(ModelComponent* currentComponent) {
        this->currentComponent = currentComponent;
    }
    ModelComponent* getCurrentComponent() const {
        return currentComponent;
    }
    void setCurrentEntity(Entity* currentEntity) {
        this->currentEntity = currentEntity;
    }
    Entity* getCurrentEntity() const {
        return currentEntity;
    }
    void setEntityCreated(Entity* entityCreated) {
        this->entityCreated = entityCreated;
    }
    Entity* getEntityCreated() const {
        return entityCreated;
    }

    void setDestinationComponent(ModelComponent* destinationComponent) {
        this->destinationComponent = destinationComponent;
    }

    ModelComponent* getDestinationComponent() const {
        return destinationComponent;
    }
private:
    SimulationEvent() {
    }
    friend class ModelSimulation;
private:
    Entity* currentEntity = nullptr;
    Entity* entityCreated = nullptr;
    Event* currentEvent = nullptr;
    ModelComponent* currentComponent = nullptr;
    ModelComponent* destinationComponent = nullptr;
    void* customObject = nullptr;
    unsigned int currentInputNumber = 0;
    unsigned int currentReplicationNumber = 0;
    double simulatedTime = 0.0;
    bool _isRunning = false;
    bool _isPaused = false;
    bool pauseRequested = false;
    bool stopRequested = false;
};

typedef void (*simulationEventHandler)(SimulationEvent*);
// for handlers that are class members (methods)
typedef std::function<void(SimulationEvent*) > simulationEventHandlerMethod;

class OnEventManager {
public:
    OnEventManager();
    virtual ~OnEventManager() = default;
public: // event listeners (handlers)
    void addOnReplicationStartHandler(simulationEventHandler EventHandler);
    void addOnReplicationStepHandler(simulationEventHandler EventHandler);
    void addOnReplicationEndHandler(simulationEventHandler EventHandler);
    void addOnProcessEventHandler(simulationEventHandler EventHandler);
    void addOnEntityCreateHandler(simulationEventHandler EventHandler);
```

```
    void addOnEntityMoveHandler(simulationEventHandler EventHandler);
    void addOnEntityRemoveHandler(simulationEventHandler EventHandler);
    void addOnSimulationStartHandler(simulationEventHandler EventHandler);
    void addOnSimulationPausedHandler(simulationEventHandler EventHandler);
    void addOnSimulationResumeHandler(simulationEventHandler EventHandler);
    void addOnSimulationEndHandler(simulationEventHandler EventHandler);
    void addOnBreakpointHandler(simulationEventHandler EventHandler);
public: // event listeners (method handlers)
    // for handlers that are class members (methods)
    template<typename Class> void addOnReplicationStartHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnReplicationStepHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnReplicationEndHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnProcessEventHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnEntityCreateHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnEntityMoveHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnEntityRemoveHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnSimulationStartHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnSimulationPausedHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnSimulationResumeHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnSimulationEndHandler(Class * object, void (Class::*function)(SimulationEvent*));
    template<typename Class> void addOnBreakpointHandler(Class * object, void (Class::*function)(SimulationEvent*));
public:
    void NotifyReplicationStartHandlers(SimulationEvent* se);
    void NotifyReplicationStepHandlers(SimulationEvent* se);
    void NotifyReplicationEndHandlers(SimulationEvent* se);
    void NotifyProcessEventHandlers(SimulationEvent* se);
    void NotifyEntityCreateHandlers(SimulationEvent* se);
    void NotifyEntityMoveHandlers(SimulationEvent* se);
    void NotifyEntityRemoveHandlers(SimulationEvent* se);
    void NotifySimulationStartHandlers(SimulationEvent* se);
    void NotifySimulationPausedHandlers(SimulationEvent* se);
    void NotifySimulationResumeHandlers(SimulationEvent* se);
    void NotifySimulationEndHandlers(SimulationEvent* se);
    void NotifyBreakpointHandlers(SimulationEvent* se);
private:
    void _NotifyHandlers(List<simulationEventHandler>* list, SimulationEvent* se);
    void _NotifyHandlerMethods(List<simulationEventHandlerMethod>* list, SimulationEvent* se);
    void _addOnHandler(List<simulationEventHandler>* list, simulationEventHandler EventHandler);
private: // events listener
    List<simulationEventHandler>* _onReplicationStartHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onReplicationStepHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onReplicationEndHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onProcessEventHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onEntityCreateHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onEntityMoveHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onEntityRemoveHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onSimulationStartHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onSimulationPausedHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onSimulationResumeHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onSimulationEndHandlers = new List<simulationEventHandler>();
    List<simulationEventHandler>* _onBreakpointHandlers = new List<simulationEventHandler>();
private: // events listener for handlers that are class members (methods)
    List<simulationEventHandlerMethod>* _onReplicationStartHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onReplicationStepHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onReplicationEndHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onProcessEventHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onEntityCreateHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onEntityMoveHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onEntityRemoveHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onSimulationStartHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onSimulationPausedHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onSimulationResumeHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onSimulationEndHandlerMethods = new List<simulationEventHandlerMethod>();
    List<simulationEventHandlerMethod>* _onBreakpointHandlerMethods = new
```

```
List<simulationEventHandlerMethod>();
};

//
// implementation for template methods
//
//  // \todo: (!!) Complicated: if handlerMethod already insert, should not insert it again. Problem to solve <...> for function
//if (_onProcessEventHandlerMethods->find(handlerMethod) == _onProcessEventHandlerMethods->list()->end())
// trying unique to solve the issue
//this->_onProcessEventHandlerMethods->list()->unique(); // does not work
//  \todo: probabily to override == operator for type simulationEventHandlerMethod
// ...
template<typename Class> void OnEventManager::addOnReplicationStartHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onReplicationStartHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnReplicationStepHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onReplicationStepHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnReplicationEndHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onReplicationEndHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnProcessEventHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onProcessEventHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnEntityCreateHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onEntityCreateHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnEntityMoveHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onEntityMoveHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnEntityRemoveHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onEntityRemoveHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnSimulationStartHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onSimulationStartHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnSimulationPausedHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onSimulationPausedHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnSimulationResumeHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onSimulationResumeHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnSimulationEndHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onSimulationEndHandlerMethods->insert(handlerMethod);
}
template<typename Class> void OnEventManager::addOnBreakpointHandler(Class * object, void (Class::*function)(SimulationEvent*)) {
    simulationEventHandlerMethod handlerMethod = std::bind(function, object, std::placeholders::_1);
    this->_onBreakpointHandlerMethods->insert(handlerMethod);
}


//namespace\\}
#endif /* ONEVENTMANAGER_H */
```

8.229 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Parser_if.h File Reference

```
#include <string>
#include "../statistics/Sampler_if.h"
```

Include dependency graph for Parser_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Parser_if

8.230 Parser_if.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Parser_if.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 23 de Agosto de 2018, 15:57
00012  */
00013
00014 #ifndef PARSER_IF_H
00015 #define PARSER_IF_H
00016
00017 #include <string>
00018 #include "../statistics/Sampler_if.h"
00019
00020 class Parser_if {
00021 public:
00022     virtual double parse(const std::string expression) = 0; // may throw exception
00023     virtual double parse(const std::string expression, bool* success, std::string* errorMessage) = 0;
       // does not throw exceptions
```

```
    virtual std::string* getErrorMessage() = 0; // to get error message in the case of thrown exception
    virtual void setSampler(Sampler_if* sampler) = 0;
    virtual Sampler_if* sampler() const = 0;
    // ...?
};

#endif /* PARSER_IF_H */
```

8.231 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/↩ Genesys++-driver.cpp File Reference

```
#include <string>
#include "Genesys++-driver.h"
```

Include dependency graph for Genesys++-driver.cpp:

8.232 Genesys++-driver.cpp

```
#include <string>
#include "Genesys++-driver.h"
//#include "../Traits.h"

//using namespace GenesysKernel;

genesyspp_driver::genesyspp_driver() {
}

genesyspp_driver::genesyspp_driver(/*GenesysKernel::*/Model* model, Sampler_if* sampler, bool throws) {
    _model = model;
    _sampler = sampler;
    throwsException = throws;
}

/*
void genesyspp_driver::scan_begin_file() {
    // IMPLEMENTED IN LEXPARSER.LL
}

void genesyspp_driver::scan_end_file() {
    // IMPLEMENTED IN LEXPARSER.LL
}


void genesyspp_driver::scan_begin_str() {
    // // IMPLEMENTED IN LEXPARSER.LL
}

void genesyspp_driver::scan_end_str() {
    // IMPLEMENTED IN LEXPARSER.LL
}
 */
int genesyspp_driver::parse_file(const std::string &f) {
    result = 0;
    file = f;
    setErrorMessage("");
    scan_begin_file();
    yy::genesyspp_parser parser(*this);
    int res = parser.parse();
    scan_end_file();
    return res;
}

```

```
int genesyspp_driver::parse_str(const std::string &str) {
    result = 0;
    str_to_parse = str;
    setErrorMessage("");
    scan_begin_str();
    yy::genesyspp_parser parser(*this);
    int res = parser.parse();
    scan_end_str();
    return res;
}

void genesyspp_driver::setResult(double value) {
    result = value;
}

double genesyspp_driver::getResult() {
    return result;
}

bool genesyspp_driver::getThrowsException() {
    return throwsException;
}

void genesyspp_driver::setThrowsException(bool throws) {
    throwsException = throws;
}

void genesyspp_driver::setErrorMessage(std::string message) {
    errorMessage = message;
}

std::string genesyspp_driver::getErrorMessage() {
    return errorMessage;
}

/*GenesysKernel::*/Model* genesyspp_driver::getModel() {
    return _model;
}

std::string genesyspp_driver::getFile() {
    return file;
}

void genesyspp_driver::setFile(std::string f) {
    file = f;
}

std::string genesyspp_driver::getStrToParse() {
    return str_to_parse;
}

void genesyspp_driver::setStrToParse(std::string str) {
    str_to_parse = str;
}

void genesyspp_driver::setSampler(Sampler_if* _sampler) {
    this->_sampler = _sampler;
}

Sampler_if* genesyspp_driver::sampler() const {
    return _sampler;
}

void
genesyspp_driver::error(const yy::location& l, const std::string& m) {
    std::string erro(m);
    erro.append("\n");
    setErrorMessage(m);
    setResult(-1);
    _model->getTracer()->trace(/*GenesysKernel::*/Util::TraceLevel::L1_errorFatal, erro);
}

void
genesyspp_driver::error(const std::string& m) {
    setErrorMessage(m);
    setResult(-1);
    _model->getTracer()->trace(/*GenesysKernel::*/Util::TraceLevel::L1_errorFatal, m);
}
```

8.233 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/↩ Genesys++-driver.h File Reference

```
#include <string>
#include <map>
#include "GenesysParser.h"
#include "../Model.h"
#include "../../util/Util.h"
#include "../../statistics/Sampler_if.h"
```

Include dependency graph for Genesys++-driver.h:

This graph shows which files directly or indirectly include this file:

Classes

- class genesyspp_driver

Macros

- #define YY_DECL yy::genesyspp_parser::symbol_type yylex (genesyspp_driver& driver)

Variables

- YY_DECL

8.233.1 Macro Definition Documentation

8.233.1.1 YY_DECL `#define YY_DECL yy::genesyspp_parser::symbol_type yylex (genesyspp_driver& driver)`

Definition at line 13 of file Genesys++-driver.h.

8.233.2 Variable Documentation

8.233.2.1 YY_DECL `YY_DECL`

Definition at line 20 of file Genesys++-driver.h.

8.234 Genesys++-driver.h

```
#ifndef GENESYSPP_DRIVER_H
#define GENESYSPP_DRIVER_H

#include <string>
#include <map>
#include "GenesysParser.h"
#include "../Model.h"
#include "../../util/Util.h"
#include "../../statistics/Sampler_if.h"
// Tell Flex the lexer's prototype ...


#define YY_DECL \
  yy::genesyspp_parser::symbol_type yylex (genesyspp_driver& driver)

/* # define YY_DECL \
//  yy::genesyspp_parser::symbol_type yylex (semantic_type* yylval, location_type* yylloc, genesyspp_driver& driver)
 */
// ... and declare it for the parser's sake.
YY_DECL;
// Conducting the whole scanning and parsing of Calc++.

//namespace GenesysKernel {

class genesyspp_driver {
public:
    genesyspp_driver();
    genesyspp_driver(/*GenesysKernel::*/Model* model, Sampler_if* sampler, bool throws = false);
    virtual ~genesyspp_driver() = default;
public:
    // Handling the scanner.
    void scan_begin_file();
    void scan_end_file();
    void scan_begin_str();
    void scan_end_str();

    // Run the parser on file F.
    // Return 0 on success.
    int parse_file(const std::string& f);
    int parse_str(const std::string& str);

    // Error handling.
    void error(const yy::location& l, const std::string& m);
    void error(const std::string& m);

    double getResult();
    void setResult(double value);

    bool getThrowsException();
    void setThrowsException(bool throws);

    void setErrorMessage(std::string message);
    std::string getErrorMessage();

    /*GenesysKernel::*/Model* getModel();
    //Sampler_if* getProbs();

    std::string getFile();
    void setFile(std::string f);

    std::string getStrToParse();
    void setStrToParse(std::string str);

public: // Sampler
    void setSampler(Sampler_if* _sampler);
    Sampler_if* sampler() const;

private:
    /*GenesysKernel::*/Model* _model;
    Sampler_if* _sampler;
private:
    double result = 0;
    std::string file;
    std::string str_to_parse;
    bool throwsException = true;
    std::string errorMessage;
```

```
};
//}
#endif // ! GENESYSPP_DRIVER_H
```

8.235 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/↩ Genesys++-scanner.cpp File Reference

```
#include <stdio.h>
#include <string.h>
#include <errno.h>
#include <stdlib.h>
#include <cerrno>
#include <climits>
#include <cstdlib>
#include <string>
#include <locale>
#include <list>
#include "Genesys++-driver.h"
#include "GenesysParser.h"
#include "obj_t.h"
#include "../../util/Util.h"
#include "../../util/List.h"
#include "../ModelElement.h"
#include "../Attribute.h"
#include "../StatisticsCollector.h"
#include "../../../plugins/elements/Variable.h"
#include "../../../plugins/elements/Queue.h"
#include "../../../plugins/elements/Formula.h"
#include "../../../plugins/elements/Resource.h"
#include "../../../plugins/elements/Set.h"
#include <unistd.h>
```

Macros

- #define YY_INT_ALIGNED short int
- #define FLEX_SCANNER
- #define YY_FLEX_MAJOR_VERSION 2
- #define YY_FLEX_MINOR_VERSION 6
- #define YY_FLEX_SUBMINOR_VERSION 4
- #define FLEX_BETA
- #define FLEXINT_H
- #define INT8_MIN (-128)
- #define INT16_MIN (-32767-1)
- #define INT32_MIN (-2147483647-1)
- #define INT8_MAX (127)
- #define INT16_MAX (32767)
- #define INT32_MAX (2147483647)
- #define UINT8_MAX (255U)
- #define UINT16_MAX (65535U)
- #define UINT32_MAX (4294967295U)
- #define SIZE_MAX (~(size_t)0)
- #define yyconst const

- #define yynoreturn
- #define YY_NULL 0
- #define YY_SC_TO_UI(c) ((YY_CHAR) (c))
- #define BEGIN (yy_start) = 1 + 2 ∗
- #define YY_START (((yy_start) - 1) / 2)
- #define YYSTATE YY_START
- #define YY_STATE_EOF(state) (YY_END_OF_BUFFER + state + 1)
- #define YY_NEW_FILE yyrestart(yyin)
- #define YY_END_OF_BUFFER_CHAR 0
- #define YY_BUF_SIZE 16384
- #define YY_STATE_BUF_SIZE ((YY_BUF_SIZE + 2) ∗ sizeof(yy_state_type))
- #define YY_TYPEDEF_YY_BUFFER_STATE
- #define YY_TYPEDEF_YY_SIZE_T
- #define EOB_ACT_CONTINUE_SCAN 0
- #define EOB_ACT_END_OF_FILE 1
- #define EOB_ACT_LAST_MATCH 2
- #define YY_LESS_LINENO(n)
- #define YY_LINENO_REWIND_TO(ptr)
- #define yyless(n)
- #define unput(c) yyunput(c, (yytext_ptr))
- #define YY_STRUCT_YY_BUFFER_STATE
- #define YY_BUFFER_NEW 0
- #define YY_BUFFER_NORMAL 1
- #define YY_BUFFER_EOF_PENDING 2
- #define YY_CURRENT_BUFFER
- #define YY_CURRENT_BUFFER_LVALUE (yy_buffer_stack)[(yy_buffer_stack_top)]
- #define YY_FLUSH_BUFFER yy_flush_buffer(YY_CURRENT_BUFFER)
- #define yy_new_buffer yy_create_buffer
- #define yy_set_interactive(is_interactive)
- #define yy_set_bol(at_bol)
- #define YY_AT_BOL() (YY_CURRENT_BUFFER_LVALUE->yy_at_bol)
- #define yywrap() (/∗CONSTCOND∗/1)
- #define YY_SKIP_YYWRAP
- #define yytext_ptr yytext
- #define YY_DO_BEFORE_ACTION
- #define YY_NUM_RULES 87
- #define YY_END_OF_BUFFER 88
- #define REJECT reject_used_but_not_detected
- #define yymore() yymore_used_but_not_detected
- #define YY_MORE_ADJ 0
- #define YY_RESTORE_YY_MORE_OFFSET
- #define yywrap() 1
- #define YY_NO_INPUT 1
- #define YY_USER_ACTION loc.columns (yyleng);
- #define INITIAL 0
- #define YY_EXTRA_TYPE void ∗
- #define YY_READ_BUF_SIZE 8192
- #define ECHO do { if (fwrite(yytext, (size_t) yyleng, 1, yyout)) {} } while (0)
- #define YY_INPUT(buf, result, max_size)
- #define yyterminate() return YY_NULL
- #define YY_START_STACK_INCR 25
- #define YY_FATAL_ERROR(msg) yy_fatal_error(msg)
- #define YY_BREAK /∗LINTED∗/break;
- #define YY_RULE_SETUP YY_USER_ACTION
- #define YY_EXIT_FAILURE 2
- #define yyless(n)
- #define YYTABLES_NAME "yytables"

Typedefs

- typedef signed char flex_int8_t
- typedef short int flex_int16_t
- typedef int flex_int32_t
- typedef unsigned char flex_uint8_t
- typedef unsigned short int flex_uint16_t
- typedef unsigned int flex_uint32_t
- typedef struct yy_buffer_state * YY_BUFFER_STATE
- typedef size_t yy_size_t
- typedef flex_uint8_t YY_CHAR
- typedef int yy_state_type

Functions

- void yyrestart (FILE *input_file)
- void yy_switch_to_buffer (YY_BUFFER_STATE new_buffer)
- YY_BUFFER_STATE yy_create_buffer (FILE *file, int size)
- void yy_delete_buffer (YY_BUFFER_STATE b)
- void yy_flush_buffer (YY_BUFFER_STATE b)
- void yypush_buffer_state (YY_BUFFER_STATE new_buffer)
- void yypop_buffer_state (void)
- static void yyensure_buffer_stack (void)
- static void yy_load_buffer_state (void)
- static void yy_init_buffer (YY_BUFFER_STATE b, FILE *file)
- YY_BUFFER_STATE yy_scan_buffer (char *base, yy_size_t size)
- YY_BUFFER_STATE yy_scan_string (const char *yy_str)
- YY_BUFFER_STATE yy_scan_bytes (const char *bytes, int len)
- void * yyalloc (yy_size_t)
- void * yyrealloc (void *, yy_size_t)
- void yyfree (void *)
- static yy_state_type yy_get_previous_state (void)
- static yy_state_type yy_try_NUL_trans (yy_state_type current_state)
- static int yy_get_next_buffer (void)
- static void yynoreturn yy_fatal_error (const char *msg)
- static int yy_init_globals (void)
- int yylex_destroy (void)
- int yyget_debug (void)
- void yyset_debug (int debug_flag)
- YY_EXTRA_TYPE yyget_extra (void)
- void yyset_extra (YY_EXTRA_TYPE user_defined)
- FILE * yyget_in (void)
- void yyset_in (FILE *_in_str)
- FILE * yyget_out (void)
- void yyset_out (FILE *_out_str)
- int yyget_leng (void)
- char * yyget_text (void)
- int yyget_lineno (void)
- void yyset_lineno (int _line_number)
- if (!(yy_init))
- while (1)

Variables

- int yyleng
- FILE * yyin = NULL
- FILE * yyout = NULL
- static size_t yy_buffer_stack_top = 0
- static size_t yy_buffer_stack_max = 0
- static YY_BUFFER_STATE * yy_buffer_stack = NULL
- static char yy_hold_char
- static int yy_n_chars
- static char * yy_c_buf_p = NULL
- static int yy_init = 0
- static int yy_start = 0
- static int yy_did_buffer_switch_on_eof
- int yylineno = 1
- char * yytext
- static const flex_int16_t yy_nxt [ ][128]
- static const flex_int16_t yy_accept [262]
- static yy_state_type yy_last_accepting_state
- static char * yy_last_accepting_cpos
- static const yy_state_type yy_NUL_trans [262]
- int yy_flex_debug = 0
- static yy::location loc
- YY_DECL
- char * yy_cp
- char * yy_bp
- int yy_act

8.235.1 Macro Definition Documentation

8.235.1.1 BEGIN `#define BEGIN (yy_start) = 1 + 2 *`

Definition at line 118 of file Genesys++-scanner.cpp.

8.235.1.2 ECHO `#define ECHO do { if (fwrite( yytext, (size_t) yyleng, 1, yyout )) {} } while (0)`

Definition at line 5128 of file Genesys++-scanner.cpp.

8.235.1.3 EOB_ACT_CONTINUE_SCAN `#define EOB_ACT_CONTINUE_SCAN 0`

Definition at line 162 of file Genesys++-scanner.cpp.

8.235.1.4 EOB_ACT_END_OF_FILE `#define EOB_ACT_END_OF_FILE 1`

Definition at line 163 of file Genesys++-scanner.cpp.

8.235.1.5 EOB_ACT_LAST_MATCH `#define EOB_ACT_LAST_MATCH 2`

Definition at line 164 of file Genesys++-scanner.cpp.

8.235.1.6 FLEX_BETA `#define FLEX_BETA`

Definition at line 14 of file Genesys++-scanner.cpp.

8.235.1.7 FLEX_SCANNER `#define FLEX_SCANNER`

Definition at line 9 of file Genesys++-scanner.cpp.

8.235.1.8 FLEXINT_H `#define FLEXINT_H`

Definition at line 30 of file Genesys++-scanner.cpp.

8.235.1.9 INITIAL `#define INITIAL 0`

Definition at line 5035 of file Genesys++-scanner.cpp.

8.235.1.10 INT16_MAX `#define INT16_MAX (32767)`

Definition at line 72 of file Genesys++-scanner.cpp.

8.235.1.11 INT16_MIN `#define INT16_MIN (-32767-1)`

Definition at line 63 of file Genesys++-scanner.cpp.

8.235.1.12 INT32_MAX `#define INT32_MAX (2147483647)`

Definition at line 75 of file Genesys++-scanner.cpp.

8.235.1.13 INT32_MIN `#define INT32_MIN (-2147483647-1)`

Definition at line 66 of file Genesys++-scanner.cpp.

8.235.1.14 INT8_MAX `#define INT8_MAX (127)`

Definition at line 69 of file Genesys++-scanner.cpp.

8.235.1.15 INT8_MIN `#define INT8_MIN (-128)`

Definition at line 60 of file Genesys++-scanner.cpp.

8.235.1.16 REJECT `#define REJECT reject_used_but_not_detected`

Definition at line 4978 of file Genesys++-scanner.cpp.

8.235.1.17 SIZE_MAX `#define SIZE_MAX (~(size_t)0)`

Definition at line 88 of file Genesys++-scanner.cpp.

8.235.1.18 UINT16_MAX `#define UINT16_MAX (65535U)`

Definition at line 81 of file Genesys++-scanner.cpp.

8.235.1.19 UINT32_MAX `#define UINT32_MAX (4294967295U)`

Definition at line 84 of file Genesys++-scanner.cpp.

8.235.1.20 UINT8_MAX `#define UINT8_MAX (255U)`

Definition at line 78 of file Genesys++-scanner.cpp.

8.235.1.21 unput `#define unput(`
`c ) yyunput( c, (yytext_ptr) )`

Definition at line 182 of file Genesys++-scanner.cpp.

8.235.1.22 YY_AT_BOL `#define YY_AT_BOL( ) (YY_CURRENT_BUFFER_LVALUE->yy_at_bol)`

Definition at line 323 of file Genesys++-scanner.cpp.

8.235.1.23 YY_BREAK `#define YY_BREAK /*LINTED*/break;`

Definition at line 5191 of file Genesys++-scanner.cpp.

8.235.1.24 YY_BUF_SIZE `#define YY_BUF_SIZE 16384`

Definition at line 140 of file Genesys++-scanner.cpp.

8.235.1.25 YY_BUFFER_EOF_PENDING `#define YY_BUFFER_EOF_PENDING 2`

Definition at line 244 of file Genesys++-scanner.cpp.

8.235.1.26 YY_BUFFER_NEW `#define YY_BUFFER_NEW 0`

Definition at line 232 of file Genesys++-scanner.cpp.

8.235.1.27 YY_BUFFER_NORMAL `#define YY_BUFFER_NORMAL 1`

Definition at line 233 of file Genesys++-scanner.cpp.

8.235.1.28 YY_CURRENT_BUFFER `#define YY_CURRENT_BUFFER`

Value:

```
( (yy_buffer_stack) \
? (yy_buffer_stack)[(yy_buffer_stack_top)] \
: NULL)
```

Definition at line 260 of file Genesys++-scanner.cpp.

8.235.1.29 YY_CURRENT_BUFFER_LVALUE `#define YY_CURRENT_BUFFER_LVALUE (yy_buffer_stack)[(yy_buffer_stack_`

Definition at line 266 of file Genesys++-scanner.cpp.

8.235.1.30 YY_DO_BEFORE_ACTION `#define YY_DO_BEFORE_ACTION`

Value:

```
(yytext_ptr) = yy_bp; \
yyleng = (int) (yy_cp - yy_bp); \
(yy_hold_char) = *yy_cp; \
*yy_cp = '\0'; \
(yy_c_buf_p) = yy_cp;
```

Definition at line 4888 of file Genesys++-scanner.cpp.

8.235.1.31 YY_END_OF_BUFFER `#define YY_END_OF_BUFFER 88`

Definition at line 4895 of file Genesys++-scanner.cpp.

8.235.1.32 YY_END_OF_BUFFER_CHAR `#define YY_END_OF_BUFFER_CHAR 0`

Definition at line 129 of file Genesys++-scanner.cpp.

8.235.1.33 YY_EXIT_FAILURE `#define YY_EXIT_FAILURE 2`

Definition at line 6593 of file Genesys++-scanner.cpp.

8.235.1.34 YY_EXTRA_TYPE `#define YY_EXTRA_TYPE void *`

Definition at line 5046 of file Genesys++-scanner.cpp.

8.235.1.35 YY_FATAL_ERROR

```
#define YY_FATAL_ERROR(
    msg ) yy_fatal_error( msg )
```

Definition at line 5166 of file Genesys++-scanner.cpp.

8.235.1.36 YY_FLEX_MAJOR_VERSION

```
#define YY_FLEX_MAJOR_VERSION 2
```

Definition at line 10 of file Genesys++-scanner.cpp.

8.235.1.37 YY_FLEX_MINOR_VERSION

```
#define YY_FLEX_MINOR_VERSION 6
```

Definition at line 11 of file Genesys++-scanner.cpp.

8.235.1.38 YY_FLEX_SUBMINOR_VERSION

```
#define YY_FLEX_SUBMINOR_VERSION 4
```

Definition at line 12 of file Genesys++-scanner.cpp.

8.235.1.39 YY_FLUSH_BUFFER

```
#define YY_FLUSH_BUFFER yy_flush_buffer( YY_CURRENT_BUFFER )
```

Definition at line 294 of file Genesys++-scanner.cpp.

8.235.1.40 YY_INPUT

```
#define YY_INPUT(
    buf,
    result,
    max_size )
```

Value:

```
    errno=0; \
    while ( (result = (int) read( fileno(yyin), buf, (yy_size_t) max_size )) < 0 ) \
    { \
        if( errno != EINTR) \
        { \
            YY_FATAL_ERROR( "input in flex scanner failed" ); \
            break; \
        } \
        errno=0; \
        clearerr(yyin); \
    }\
\
```

Definition at line 5135 of file Genesys++-scanner.cpp.

8.235.1.41 YY_INT_ALIGNED `#define YY_INT_ALIGNED short int`

Definition at line 5 of file Genesys++-scanner.cpp.

8.235.1.42 YY_LESS_LINENO `#define YY_LESS_LINENO( n )`

Definition at line 166 of file Genesys++-scanner.cpp.

8.235.1.43 YY_LINENO_REWIND_TO `#define YY_LINENO_REWIND_TO( ptr )`

Definition at line 167 of file Genesys++-scanner.cpp.

8.235.1.44 YY_MORE_ADJ `#define YY_MORE_ADJ 0`

Definition at line 4980 of file Genesys++-scanner.cpp.

8.235.1.45 yy_new_buffer `#define yy_new_buffer yy_create_buffer`

Definition at line 304 of file Genesys++-scanner.cpp.

8.235.1.46 YY_NEW_FILE `#define YY_NEW_FILE yyrestart( yyin )`

Definition at line 128 of file Genesys++-scanner.cpp.

8.235.1.47 YY_NO_INPUT `#define YY_NO_INPUT 1`

Definition at line 5028 of file Genesys++-scanner.cpp.

8.235.1.48 YY_NULL `#define YY_NULL 0`

Definition at line 107 of file Genesys++-scanner.cpp.

8.235.1.49 YY_NUM_RULES `#define YY_NUM_RULES 87`

Definition at line 4894 of file Genesys++-scanner.cpp.

8.235.1.50 YY_READ_BUF_SIZE `#define YY_READ_BUF_SIZE 8192`

Definition at line 5119 of file Genesys++-scanner.cpp.

8.235.1.51 YY_RESTORE_YY_MORE_OFFSET `#define YY_RESTORE_YY_MORE_OFFSET`

Definition at line 4981 of file Genesys++-scanner.cpp.

8.235.1.52 YY_RULE_SETUP `#define YY_RULE_SETUP YY_USER_ACTION`

Definition at line 5194 of file Genesys++-scanner.cpp.

8.235.1.53 YY_SC_TO_UI `#define YY_SC_TO_UI( c ) ((YY_CHAR) (c))`

Definition at line 112 of file Genesys++-scanner.cpp.

8.235.1.54 yy_set_bol `#define yy_set_bol( at_bol )`

Value:

```
{ \
if ( ! YY_CURRENT_BUFFER ){\
    yyensure_buffer_stack (); \
    YY_CURRENT_BUFFER_LVALUE =    \
        yy_create_buffer( yyin, YY_BUF_SIZE ); \
} \
YY_CURRENT_BUFFER_LVALUE->yy_at_bol = at_bol; \
}
```

Definition at line 314 of file Genesys++-scanner.cpp.

8.235.1.55 yy_set_interactive

```
#define yy_set_interactive(
            is_interactive )
```

Value:

```
{ \
if ( ! YY_CURRENT_BUFFER ){ \
    yyensure_buffer_stack (); \
    YY_CURRENT_BUFFER_LVALUE =    \
        yy_create_buffer( yyin, YY_BUF_SIZE ); \
} \
YY_CURRENT_BUFFER_LVALUE->yy_is_interactive = is_interactive; \
}
```

Definition at line 305 of file Genesys++-scanner.cpp.

8.235.1.56 YY_SKIP_YYWRAP

```
#define YY_SKIP_YYWRAP
```

Definition at line 328 of file Genesys++-scanner.cpp.

8.235.1.57 YY_START

```
#define YY_START (((yy_start) - 1) / 2)
```

Definition at line 123 of file Genesys++-scanner.cpp.

8.235.1.58 YY_START_STACK_INCR

```
#define YY_START_STACK_INCR 25
```

Definition at line 5161 of file Genesys++-scanner.cpp.

8.235.1.59 YY_STATE_BUF_SIZE

```
#define YY_STATE_BUF_SIZE ((YY_BUF_SIZE + 2) * sizeof(yy_state_type))
```

Definition at line 146 of file Genesys++-scanner.cpp.

8.235.1.60 YY_STATE_EOF

```
#define YY_STATE_EOF(
            state ) (YY_END_OF_BUFFER + state + 1)
```

Definition at line 126 of file Genesys++-scanner.cpp.

8.235.1.61 YY_STRUCT_YY_BUFFER_STATE

```
#define YY_STRUCT_YY_BUFFER_STATE
```

Definition at line 185 of file Genesys++-scanner.cpp.

8.235.1.62 YY_TYPEDEF_YY_BUFFER_STATE `#define YY_TYPEDEF_YY_BUFFER_STATE`

Definition at line 149 of file Genesys++-scanner.cpp.

8.235.1.63 YY_TYPEDEF_YY_SIZE_T `#define YY_TYPEDEF_YY_SIZE_T`

Definition at line 154 of file Genesys++-scanner.cpp.

8.235.1.64 YY_USER_ACTION `#define YY_USER_ACTION loc.columns (yyleng);`

Definition at line 5031 of file Genesys++-scanner.cpp.

8.235.1.65 yyconst `#define yyconst const`

Definition at line 98 of file Genesys++-scanner.cpp.

8.235.1.66 yyless [1/2] `#define yyless(`
`n )`

Value:

```
do \
    { \
    /* Undo effects of setting up yytext. */ \
    int yyless_macro_arg = (n); \
    YY_LESS_LINENO(yyless_macro_arg);\
    *yy_cp = (yy_hold_char); \
    YY_RESTORE_YY_MORE_OFFSET \
    (yy_c_buf_p) = yy_cp = yy_bp + yyless_macro_arg - YY_MORE_ADJ; \
    YY_DO_BEFORE_ACTION; /* set up yytext again */ \
    } \
while ( 0 )
```

Definition at line 6605 of file Genesys++-scanner.cpp.

8.235.1.67 yyless [2/2] `#define yyless(`
`n )`

Value:

```
do \
    { \
    /* Undo effects of setting up yytext. */ \
    int yyless_macro_arg = (n); \
    YY_LESS_LINENO(yyless_macro_arg);\
    yytext[yyleng] = (yy_hold_char); \
    (yy_c_buf_p) = yytext + yyless_macro_arg; \
    (yy_hold_char) = *(yy_c_buf_p); \
    *(yy_c_buf_p) = '\0'; \
    yyleng = yyless_macro_arg; \
    } \
while ( 0 )
```

Definition at line 6605 of file Genesys++-scanner.cpp.

8.235.1.68 yymore `#define yymore( ) yymore_used_but_not_detected`

Definition at line 4979 of file Genesys++-scanner.cpp.

8.235.1.69 yynoreturn `#define yynoreturn`

Definition at line 103 of file Genesys++-scanner.cpp.

8.235.1.70 YYSTATE `#define YYSTATE YY_START`

Definition at line 124 of file Genesys++-scanner.cpp.

8.235.1.71 YYTABLES_NAME `#define YYTABLES_NAME "yytables"`

Definition at line 6797 of file Genesys++-scanner.cpp.

8.235.1.72 yyterminate `#define yyterminate( ) return YY_NULL`

Definition at line 5156 of file Genesys++-scanner.cpp.

8.235.1.73 yytext_ptr `#define yytext_ptr yytext`

Definition at line 342 of file Genesys++-scanner.cpp.

8.235.1.74 yywrap [1/2] `#define yywrap( ) (/*CONSTCOND*/1)`

begin_Includes_plugins begin_Includes:Variable end_Includes:Variable begin_Includes:Queue end_Includes↩
:Queue begin_Includes:Formula end_Includes:Formula begin_Includes:Resource end_Includes:Resource begin↩
_Includes:Set end_Includes:Set end_Includes_plugins

Definition at line 5023 of file Genesys++-scanner.cpp.

8.235.1.75 yywrap [2/2] `#define yywrap( ) 1`

begin_Includes_plugins begin_Includes:Variable end_Includes:Variable begin_Includes:Queue end_Includes↩
:Queue begin_Includes:Formula end_Includes:Formula begin_Includes:Resource end_Includes:Resource begin↩
_Includes:Set end_Includes:Set end_Includes_plugins

Definition at line 5023 of file Genesys++-scanner.cpp.

8.235.2 Typedef Documentation

8.235.2.1 flex_int16_t `typedef short int flex_int16_t`

Definition at line 52 of file Genesys++-scanner.cpp.

8.235.2.2 flex_int32_t `typedef int flex_int32_t`

Definition at line 53 of file Genesys++-scanner.cpp.

8.235.2.3 flex_int8_t `typedef signed char flex_int8_t`

Definition at line 51 of file Genesys++-scanner.cpp.

8.235.2.4 flex_uint16_t `typedef unsigned short int flex_uint16_t`

Definition at line 55 of file Genesys++-scanner.cpp.

8.235.2.5 flex_uint32_t `typedef unsigned int flex_uint32_t`

Definition at line 56 of file Genesys++-scanner.cpp.

8.235.2.6 flex_uint8_t `typedef unsigned char flex_uint8_t`

Definition at line 54 of file Genesys++-scanner.cpp.

8.235.2.7 YY_BUFFER_STATE `typedef struct yy_buffer_state* YY_BUFFER_STATE`

Definition at line 150 of file Genesys++-scanner.cpp.

8.235.2.8 YY_CHAR `typedef flex_uint8_t YY_CHAR`

Definition at line 329 of file Genesys++-scanner.cpp.

8.235.2.9 yy_size_t `typedef size_t yy_size_t`

Definition at line 155 of file Genesys++-scanner.cpp.

8.235.2.10 yy_state_type `typedef int yy_state_type`

Definition at line 333 of file Genesys++-scanner.cpp.

8.235.3 Function Documentation

8.235.3.1 if() `if ( ! yy_init )`

Definition at line 5205 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

8.235.3.2 while() `while ( 1 )`

begin_Lexical:Resource

end_Lexical:Resource

begin_Lexical:Queue

end_Lexical:Queue

begin_Lexical:Set

end_Lexical:Set

begin_Lexical:EntityGroup

end_Lexical:EntityGroup

begin_LexicalLiterals:Variable

end_LexicalLiterals:Variable

begin_LexicalLiterals:Formula

end_LexicalLiterals:Formula

begin_LexicalLiterals:Queue

end_LexicalLiterals:Queue

begin_LexicalLiterals:Resource

end_LexicalLiterals:Resource

begin_LexicalLiterals:Set

end_LexicalLiterals:Set

Definition at line 5244 of file Genesys++-scanner.cpp.

8.235.3.3 yy_create_buffer() `YY_BUFFER_STATE yy_create_buffer ( FILE * file, int size )`

Allocate and initialize an input buffer state.

Parameters

file	A readable stream.
size	The character buffer size in bytes. When in doubt, use `YY_BUF_SIZE`.

Returns

the allocated buffer state.

Definition at line 6310 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.4 yy_delete_buffer()

```
void yy_delete_buffer (
    YY_BUFFER_STATE b )
```

Destroy the buffer.

Parameters

b	a buffer created with yy_create_buffer()

Definition at line 6338 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.5 yy_fatal_error() `static void yynoreturn yy_fatal_error ( const char * msg ) [static]`

Definition at line 6596 of file Genesys++-scanner.cpp.

8.235.3.6 yy_flush_buffer() `void yy_flush_buffer ( YY_BUFFER_STATE b )`

Discard all buffered characters. On the next scan, YY_INPUT will be called.

Parameters

b	the buffer state to be flushed, usually `YY_CURRENT_BUFFER`.

Definition at line 6385 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.7 yy_get_next_buffer()

```
static int yy_get_next_buffer (
    void ) [static]
```

Definition at line 5977 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.8 yy_get_previous_state()

```
static yy_state_type yy_get_previous_state (
    void ) [static]
```

Definition at line 6115 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.9 yy_init_buffer()

```
static void yy_init_buffer (
    YY_BUFFER_STATE b,
    FILE * file )  [static]
```

Definition at line 6357 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.10 yy_init_globals()

```
static int yy_init_globals (
    void )  [static]
```

Definition at line 6699 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.11 yy_load_buffer_state()

```
static void yy_load_buffer_state (
    void ) [static]
```

Definition at line 6296 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.12 yy_scan_buffer()

```
YY_BUFFER_STATE yy_scan_buffer (
    char * base,
    yy_size_t size )
```

Setup the input buffer state to scan directly from a user-specified character buffer.

Parameters

base	the character buffer
size	the size in bytes of the character buffer

Returns

the newly allocated buffer state object.

Definition at line 6512 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.13 yy_scan_bytes()

```
YY_BUFFER_STATE yy_scan_bytes (
    const char * yybytes,
    int _yybytes_len )
```

Setup the input buffer state to scan the given bytes. The next call to yylex() will scan from a *copy* of *bytes*.

Parameters

yybytes	the byte buffer to scan
_yybytes_len	the number of bytes in the buffer pointed to by *bytes*.

Returns

the newly allocated buffer state object.

Definition at line 6562 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.14 yy_scan_string()

```
YY_BUFFER_STATE yy_scan_string (
    const char * yystr )
```

Setup the input buffer state to scan a string. The next call to yylex() will scan from a *copy* of *str*.

Parameters

yystr	a NUL-terminated string to scan

Returns

the newly allocated buffer state object.

Note

If you want to scan bytes that may contain NUL values, then use yy_scan_bytes() instead.

Definition at line 6549 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.15 yy_switch_to_buffer() `void yy_switch_to_buffer ( YY_BUFFER_STATE new_buffer )`

Switch to a different input buffer.

Parameters

new_buffer	The new input buffer.

Definition at line 6265 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.16 yy_try_NUL_trans() `static yy_state_type yy_try_NUL_trans ( yy_state_type current_state ) [static]`

Definition at line 6145 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.17 yyalloc() `void * yyalloc ( yy_size_t size )`

Definition at line 6774 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.18 yyensure_buffer_stack()

```
static void yyensure_buffer_stack (
    void ) [static]
```

Definition at line 6463 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.19 yyfree()

```
void yyfree (
    void * ptr )
```

Definition at line 6792 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.20 yyget_debug()

```
int yyget_debug (
    void )
```

Definition at line 6689 of file Genesys++-scanner.cpp.

8.235.3.21 yyget_extra()

```
YY_EXTRA_TYPE yyget_extra (
void )
```

8.235.3.22 yyget_in()

```
FILE * yyget_in (
void )
```

Get the input stream.

Definition at line 6633 of file Genesys++-scanner.cpp.

8.235.3.23 yyget_leng()

```
int yyget_leng (
void )
```

Get the length of the current token.

Definition at line 6649 of file Genesys++-scanner.cpp.

8.235.3.24 yyget_lineno()

```
int yyget_lineno (
void )
```

Get the current line number.

Definition at line 6624 of file Genesys++-scanner.cpp.

8.235.3.25 yyget_out()

```
FILE * yyget_out (
void )
```

Get the output stream.

Definition at line 6641 of file Genesys++-scanner.cpp.

8.235.3.26 yyget_text()

```
char * yyget_text (
void )
```

Get the current token.

Definition at line 6658 of file Genesys++-scanner.cpp.

8.235.3.27 yylex_destroy() `int yylex_destroy ( void )`

Definition at line 6728 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

8.235.3.28 yypop_buffer_state() `void yypop_buffer_state ( void )`

Removes and deletes the top of the stack, if present. The next element becomes the new top.

Definition at line 6444 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.29 yypush_buffer_state() `void yypush_buffer_state ( YY_BUFFER_STATE new_buffer )`

Pushes the new state onto the stack. The new state becomes the current state. This function will allocate the stack if necessary.

Parameters

new_buffer	The new state.

Definition at line 6414 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

8.235.3.30 yyrealloc()

```
void * yyrealloc (
    void * ptr,
    yy_size_t size )
```

Definition at line 6779 of file Genesys++-scanner.cpp.

Here is the caller graph for this function:

8.235.3.31 yyrestart()

```
void yyrestart (
    FILE * input_file )
```

Immediately switch to a different input stream.

Parameters

input_file	A readable stream.

Note

This function does not reset the start condition to `INITIAL` .

Definition at line 6248 of file Genesys++-scanner.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.235.3.32 yyset_debug()

```
void yyset_debug (
    int debug_flag )
```

Definition at line 6694 of file Genesys++-scanner.cpp.

8.235.3.33 yyset_extra()

```
void yyset_extra (
    YY_EXTRA_TYPE user_defined )
```

8.235.3.34 yyset_in()

```
void yyset_in (
    FILE * _in_str )
```

Set the input stream. This does not discard the current input buffer.

Parameters

_in_str	A readable stream.

See also

yy_switch_to_buffer

Definition at line 6679 of file Genesys++-scanner.cpp.

8.235.3.35 yyset_lineno() `void yyset_lineno ( int _line_number )`

Set the current line number.

Parameters

_line_number	line number

Definition at line 6667 of file Genesys++-scanner.cpp.

8.235.3.36 yyset_out() `void yyset_out ( FILE * _out_str )`

Definition at line 6684 of file Genesys++-scanner.cpp.

8.235.4 Variable Documentation

8.235.4.1 loc `yy::location loc [static]`

Definition at line 5026 of file Genesys++-scanner.cpp.

8.235.4.2 yy_accept `const flex_int16_t yy_accept[262] [static]`

Definition at line 4903 of file Genesys++-scanner.cpp.

8.235.4.3 yy_act `int yy_act`

Definition at line 5203 of file Genesys++-scanner.cpp.

8.235.4.4 yy_bp `char * yy_bp`

Definition at line 5202 of file Genesys++-scanner.cpp.

8.235.4.5 yy_buffer_stack `YY_BUFFER_STATE* yy_buffer_stack = NULL [static]`

Stack as an array.

Definition at line 252 of file Genesys++-scanner.cpp.

8.235.4.6 yy_buffer_stack_max `size_t yy_buffer_stack_max = 0 [static]`

capacity of stack.

Definition at line 251 of file Genesys++-scanner.cpp.

8.235.4.7 yy_buffer_stack_top `size_t yy_buffer_stack_top = 0 [static]`

index of top of stack.

Definition at line 250 of file Genesys++-scanner.cpp.

8.235.4.8 yy_c_buf_p `char* yy_c_buf_p = NULL [static]`

Definition at line 274 of file Genesys++-scanner.cpp.

8.235.4.9 yy_cp `char* yy_cp`

Definition at line 5202 of file Genesys++-scanner.cpp.

8.235.4.10 YY_DECL `YY_DECL`

Initial value:

```
{
    yy_state_type yy_current_state
```

The main scanner function which does all the work.

Definition at line 5199 of file Genesys++-scanner.cpp.

8.235.4.11 yy_did_buffer_switch_on_eof `int yy_did_buffer_switch_on_eof [static]`

Definition at line 281 of file Genesys++-scanner.cpp.

8.235.4.12 yy_flex_debug `int yy_flex_debug = 0`

Definition at line 4973 of file Genesys++-scanner.cpp.

8.235.4.13 yy_hold_char `char yy_hold_char [static]`

Definition at line 269 of file Genesys++-scanner.cpp.

8.235.4.14 yy_init `int yy_init = 0 [static]`

Definition at line 275 of file Genesys++-scanner.cpp.

8.235.4.15 yy_last_accepting_cpos `char* yy_last_accepting_cpos [static]`

Definition at line 4937 of file Genesys++-scanner.cpp.

8.235.4.16 yy_last_accepting_state `yy_state_type yy_last_accepting_state [static]`

Definition at line 4936 of file Genesys++-scanner.cpp.

8.235.4.17 yy_n_chars `int yy_n_chars [static]`

Definition at line 270 of file Genesys++-scanner.cpp.

8.235.4.18 yy_NUL_trans `const yy_state_type yy_NUL_trans[262] [static]`

Definition at line 4939 of file Genesys++-scanner.cpp.

8.235.4.19 yy_nxt `const flex_int16_t yy_nxt[][128] [static]`

Definition at line 344 of file Genesys++-scanner.cpp.

8.235.4.20 yy_start `int yy_start = 0 [static]`

Definition at line 276 of file Genesys++-scanner.cpp.

8.235.4.21 yyin `FILE * yyin = NULL`

Definition at line 331 of file Genesys++-scanner.cpp.

8.235.4.22 yyleng `int yyleng`

Definition at line 271 of file Genesys++-scanner.cpp.

8.235.4.23 yylineno `int yylineno = 1`

Definition at line 336 of file Genesys++-scanner.cpp.

8.235.4.24 yyout `FILE * yyout = NULL`

Definition at line 160 of file Genesys++-scanner.cpp.

8.235.4.25 yytext `char * yytext`

Definition at line 4982 of file Genesys++-scanner.cpp.

8.236 Genesys++-scanner.cpp

```
#line 2 "../Genesys++-scanner.cpp"

#line 4 "../Genesys++-scanner.cpp"

#define  YY_INT_ALIGNED short int

/* A lexical scanner generated by flex */

#define FLEX_SCANNER
#define YY_FLEX_MAJOR_VERSION 2
#define YY_FLEX_MINOR_VERSION 6
#define YY_FLEX_SUBMINOR_VERSION 4
#if YY_FLEX_SUBMINOR_VERSION > 0
#define FLEX_BETA
#endif

/* First, we deal with  platform-specific or compiler-specific issues. */

/* begin standard C headers. */
#include <stdio.h>
#include <string.h>
#include <errno.h>
#include <stdlib.h>

/* end standard C headers. */

/* flex integer type definitions */

#ifndef FLEXINT_H
#define FLEXINT_H

/* C99 systems have <inttypes.h>. Non-C99 systems may or may not. */

#if defined (__STDC_VERSION__) && __STDC_VERSION__ >= 199901L

/* C99 says to define __STDC_LIMIT_MACROS before including stdint.h,
 * if you want the limit (max/min) macros for int types.
 */
#ifndef __STDC_LIMIT_MACROS
#define __STDC_LIMIT_MACROS 1
#endif

#include <inttypes.h>
typedef int8_t flex_int8_t;
typedef uint8_t flex_uint8_t;
typedef int16_t flex_int16_t;
typedef uint16_t flex_uint16_t;
typedef int32_t flex_int32_t;
typedef uint32_t flex_uint32_t;
#else
typedef signed char flex_int8_t;
typedef short int flex_int16_t;
typedef int flex_int32_t;
typedef unsigned char flex_uint8_t;
typedef unsigned short int flex_uint16_t;
typedef unsigned int flex_uint32_t;

/* Limits of integral types. */
#ifndef INT8_MIN
#define INT8_MIN               (-128)
#endif
#ifndef INT16_MIN
#define INT16_MIN              (-32767-1)
#endif
#ifndef INT32_MIN
#define INT32_MIN              (-2147483647-1)
#endif
#ifndef INT8_MAX
#define INT8_MAX               (127)
#endif
#ifndef INT16_MAX
#define INT16_MAX              (32767)
#endif
#ifndef INT32_MAX
#define INT32_MAX              (2147483647)
#endif
#ifndef UINT8_MAX
#define UINT8_MAX              (255U)
#endif
#ifndef UINT16_MAX
#define UINT16_MAX             (65535U)
#endif
#ifndef UINT32_MAX
#define UINT32_MAX             (4294967295U)
#endif
```

```

#ifndef SIZE_MAX
#define SIZE_MAX               (~(size_t)0)
#endif

#endif /* ! C99 */

#endif /* ! FLEXINT_H */

/* begin standard C++ headers. */

/* TODO: this is always defined, so inline it */
#define yyconst const

#if defined(__GNUC__) && __GNUC__ >= 3
#define yynoreturn __attribute__((__noreturn__))
#else
#define yynoreturn
#endif

/* Returned upon end-of-file. */
#define YY_NULL 0

/* Promotes a possibly negative, possibly signed char to an
 *   integer in range [0..255] for use as an array index.
 */
#define YY_SC_TO_UI(c) ((YY_CHAR) (c))

/* Enter a start condition.  This macro really ought to take a parameter,
 * but we do it the disgusting crufty way forced on us by the ()-less
 * definition of BEGIN.
 */
#define BEGIN (yy_start) = 1 + 2 *
/* Translate the current start state into a value that can be later handed
 * to BEGIN to return to the state.  The YYSTATE alias is for lex
 * compatibility.
 */
#define YY_START (((yy_start) - 1) / 2)
#define YYSTATE YY_START
/* Action number for EOF rule of a given start state. */
#define YY_STATE_EOF(state) (YY_END_OF_BUFFER + state + 1)
/* Special action meaning "start processing a new file". */
#define YY_NEW_FILE yyrestart( yyin  )
#define YY_END_OF_BUFFER_CHAR 0

/* Size of default input buffer. */
#ifndef YY_BUF_SIZE
#ifdef __ia64__
/* On IA-64, the buffer size is 16k, not 8k.
 * Moreover, YY_BUF_SIZE is 2*YY_READ_BUF_SIZE in the general case.
 * Ditto for the __ia64__ case accordingly.
 */
#define YY_BUF_SIZE 32768
#else
#define YY_BUF_SIZE 16384
#endif /* __ia64__ */
#endif

/* The state buf must be large enough to hold one state per character in the main buffer.
 */
#define YY_STATE_BUF_SIZE   ((YY_BUF_SIZE + 2) * sizeof(yy_state_type))

#ifndef YY_TYPEDEF_YY_BUFFER_STATE
#define YY_TYPEDEF_YY_BUFFER_STATE
typedef struct yy_buffer_state *YY_BUFFER_STATE;
#endif

#ifndef YY_TYPEDEF_YY_SIZE_T
#define YY_TYPEDEF_YY_SIZE_T
typedef size_t yy_size_t;
#endif

extern int yyleng;

extern FILE *yyin, *yyout;

#define EOB_ACT_CONTINUE_SCAN 0
#define EOB_ACT_END_OF_FILE 1
#define EOB_ACT_LAST_MATCH 2

    #define YY_LESS_LINENO(n)
    #define YY_LINENO_REWIND_TO(ptr)

/* Return all but the first "n" matched characters back to the input stream. */
#define yyless(n) \
    do \
        { \
```

```
        /* Undo effects of setting up yytext. */ \
        int yyless_macro_arg = (n); \
        YY_LESS_LINENO(yyless_macro_arg);\
        *yy_cp = (yy_hold_char); \
        YY_RESTORE_YY_MORE_OFFSET \
        (yy_c_buf_p) = yy_cp = yy_bp + yyless_macro_arg - YY_MORE_ADJ; \
        YY_DO_BEFORE_ACTION; /* set up yytext again */ \
        } \
    while ( 0 )
#define unput(c) yyunput( c, (yytext_ptr)  )

#ifndef YY_STRUCT_YY_BUFFER_STATE
#define YY_STRUCT_YY_BUFFER_STATE
struct yy_buffer_state
    {
    FILE *yy_input_file;

    char *yy_ch_buf;        /* input buffer */
    char *yy_buf_pos;       /* current position in input buffer */

    /* Size of input buffer in bytes, not including room for EOB
     * characters.
     */
    int yy_buf_size;

    /* Number of characters read into yy_ch_buf, not including EOB
     * characters.
     */
    int yy_n_chars;

    /* Whether we "own" the buffer - i.e., we know we created it,
     * and can realloc() it to grow it, and should free() it to
     * delete it.
     */
    int yy_is_our_buffer;

    /* Whether this is an "interactive" input source; if so, and
     * if we're using stdio for input, then we want to use getc()
     * instead of fread(), to make sure we stop fetching input after
     * each newline.
     */
    int yy_is_interactive;

    /* Whether we're considered to be at the beginning of a line.
     * If so, '^' rules will be active on the next match, otherwise
     * not.
     */
    int yy_at_bol;

    int yy_bs_lineno;
    int yy_bs_column;
    /* Whether to try to fill the input buffer when we reach the
     * end of it.
     */
    int yy_fill_buffer;

    int yy_buffer_status;

#define YY_BUFFER_NEW 0
#define YY_BUFFER_NORMAL 1
    /* When an EOF's been seen but there's still some text to process
     * then we mark the buffer as YY_EOF_PENDING, to indicate that we
     * shouldn't try reading from the input source any more.  We might
     * still have a bunch of tokens to match, though, because of
     * possible backing-up.
     *
     * When we actually see the EOF, we change the status to "new"
     * (via yyrestart()), so that the user can continue scanning by
     * just pointing yyin at a new input file.
     */
#define YY_BUFFER_EOF_PENDING 2

    };
#endif /* !YY_STRUCT_YY_BUFFER_STATE */

/* Stack of input buffers. */
static size_t yy_buffer_stack_top = 0;
static size_t yy_buffer_stack_max = 0;
static YY_BUFFER_STATE * yy_buffer_stack = NULL;
/* We provide macros for accessing buffer states in case in the
 * future we want to put the buffer states in a more general
 * "scanner state".
 *
 * Returns the top of the stack, or NULL.
 */
#define YY_CURRENT_BUFFER ( (yy_buffer_stack) \
                          ? (yy_buffer_stack)[(yy_buffer_stack_top)] \
```

```
                          : NULL)
/* Same as previous macro, but useful when we know that the buffer stack is not
 * NULL or when we need an lvalue. For internal use only.
 */
#define YY_CURRENT_BUFFER_LVALUE (yy_buffer_stack)[(yy_buffer_stack_top)]

/* yy_hold_char holds the character lost when yytext is formed. */
static char yy_hold_char;
static int yy_n_chars;        /* number of characters read into yy_ch_buf */
int yyleng;

/* Points to current character in buffer. */
static char *yy_c_buf_p = NULL;
static int yy_init = 0;      /* whether we need to initialize */
static int yy_start = 0;    /* start state number */

/* Flag which is used to allow yywrap()'s to do buffer switches
 * instead of setting up a fresh yyin.  A bit of a hack ...
 */
static int yy_did_buffer_switch_on_eof;

void yyrestart ( FILE *input_file  );
void yy_switch_to_buffer ( YY_BUFFER_STATE new_buffer  );
YY_BUFFER_STATE yy_create_buffer ( FILE *file, int size  );
void yy_delete_buffer ( YY_BUFFER_STATE b  );
void yy_flush_buffer ( YY_BUFFER_STATE b  );
void yypush_buffer_state ( YY_BUFFER_STATE new_buffer  );
void yypop_buffer_state ( void );

static void yyensure_buffer_stack ( void );
static void yy_load_buffer_state ( void );
static void yy_init_buffer ( YY_BUFFER_STATE b, FILE *file  );
#define YY_FLUSH_BUFFER yy_flush_buffer( YY_CURRENT_BUFFER )

YY_BUFFER_STATE yy_scan_buffer ( char *base, yy_size_t size  );
YY_BUFFER_STATE yy_scan_string ( const char *yy_str  );
YY_BUFFER_STATE yy_scan_bytes ( const char *bytes, int len  );

void *yyalloc ( yy_size_t  );
void *yyrealloc ( void *, yy_size_t  );
void yyfree ( void *  );

#define yy_new_buffer yy_create_buffer
#define yy_set_interactive(is_interactive) \
    { \
    if ( ! YY_CURRENT_BUFFER ){ \
        yyensure_buffer_stack (); \
        YY_CURRENT_BUFFER_LVALUE =    \
            yy_create_buffer( yyin, YY_BUF_SIZE ); \
    } \
    YY_CURRENT_BUFFER_LVALUE->yy_is_interactive = is_interactive; \
    }
#define yy_set_bol(at_bol) \
    { \
    if ( ! YY_CURRENT_BUFFER ){\
        yyensure_buffer_stack (); \
        YY_CURRENT_BUFFER_LVALUE =    \
            yy_create_buffer( yyin, YY_BUF_SIZE ); \
    } \
    YY_CURRENT_BUFFER_LVALUE->yy_at_bol = at_bol; \
    }
#define YY_AT_BOL() (YY_CURRENT_BUFFER_LVALUE->yy_at_bol)

/* Begin user sect3 */

#define yywrap() (/*CONSTCOND*/1)
#define YY_SKIP_YYWRAP
typedef flex_uint8_t YY_CHAR;

FILE *yyin = NULL, *yyout = NULL;

typedef int yy_state_type;

extern int yylineno;
int yylineno = 1;

extern char *yytext;
#ifdef yytext_ptr
#undef yytext_ptr
#endif
#define yytext_ptr yytext

static const flex_int16_t yy_nxt[][128] =
    {
    {
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
```

```
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,

        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0
    },

    {
        3,    4,    4,    4,    4,    4,    4,    4,    4,    5,
        6,    4,    4,    4,    4,    4,    4,    4,    4,    4,
        4,    4,    4,    4,    4,    4,    4,    4,    4,    4,
        4,    4,    5,    4,    4,    4,    4,    4,    4,    4,
        7,    8,    9,   10,   11,   12,   13,   14,   15,   16,
       16,   16,   16,   16,   16,   16,   16,   16,    4,    4,
       17,   18,   19,    4,    4,   20,   21,   22,   23,   24,

       25,   26,   13,   27,   13,   13,   28,   29,   30,   31,
       13,   13,   32,   33,   34,   35,   36,   37,   38,   13,
       13,   39,    4,   40,   41,   13,    4,   20,   21,   22,
       23,   24,   25,   26,   13,   27,   13,   13,   28,   29,
       30,   31,   13,   13,   32,   33,   42,   35,   36,   37,
       38,   13,   13,    4,    4,    4,    4,    4
    },

    {
        3,    4,    4,    4,    4,    4,    4,    4,    4,    5,
        6,    4,    4,    4,    4,    4,    4,    4,    4,    4,
        4,    4,    4,    4,    4,    4,    4,    4,    4,    4,
        4,    4,    5,    4,    4,    4,    4,    4,    4,    4,

        7,    8,    9,   10,   11,   12,   13,   14,   15,   16,
       16,   16,   16,   16,   16,   16,   16,   16,    4,    4,
       17,   18,   19,    4,    4,   20,   21,   22,   23,   24,
       25,   26,   13,   27,   13,   13,   28,   29,   30,   31,
       13,   13,   32,   33,   34,   35,   36,   37,   38,   13,
       13,   39,    4,   40,   41,   13,    4,   20,   21,   22,
       23,   24,   25,   26,   13,   27,   13,   13,   28,   29,
       30,   31,   13,   13,   32,   33,   42,   35,   36,   37,
       38,   13,   13,    4,    4,    4,    4,    4
    },

    {
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,

       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,

       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3,
       -3,   -3,   -3,   -3,   -3,   -3,   -3,   -3
    },

    {
        3,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,

       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4,
       -4,   -4,   -4,   -4,   -4,   -4,   -4,   -4
    },

    {
        3,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
       -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
       -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
```

```
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,

        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5,
        -5,   -5,   -5,   -5,   -5,   -5,   -5,   -5
    },

    {
         3,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,

        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,
        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6,

        -6,   -6,   -6,   -6,   -6,   -6,   -6,   -6
    },

    {
         3,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,

        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7,
        -7,   -7,   -7,   -7,   -7,   -7,   -7,   -7
    },

    {
         3,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,

        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8,
        -8,   -8,   -8,   -8,   -8,   -8,   -8,   -8
    },

    {
         3,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,

        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9,
        -9,   -9,   -9,   -9,   -9,   -9,   -9,   -9

    },

    {
         3,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
       -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
       -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
```

```
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,

      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10,
      -10,  -10,  -10,  -10,  -10,  -10,  -10,  -10
    },

    {
        3,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,

      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11,
      -11,  -11,  -11,  -11,  -11,  -11,  -11,  -11
    },

    {
        3,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,

      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12,
      -12,  -12,  -12,  -12,  -12,  -12,  -12,  -12
    },

    {
        3,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,

      -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,
      -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,
      -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,  -13,
      -13,  -13,  -13,  -13,  -13,  -13,   43,  -13,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -13,  -13,
      -13,  -13,  -13,  -13,  -13,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -13,  -13,  -13,  -13,   43,  -13,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -13,  -13,  -13,  -13,  -13
    },

    {
        3,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,

      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14,
      -14,  -14,  -14,  -14,  -14,  -14,  -14,  -14
    },

    {
        3,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,
      -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,
      -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,
      -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,  -15,
```

```
      -15,  -15,  -15,  -15,  -15,  -15,   44,  -15,   45,   45,

       45,   45,   45,   45,   45,   45,   45,   45,  -15,  -15,
      -15,  -15,  -15,  -15,  -15,   43,   43,   43,   43,   46,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   47,   43,
       43,  -15,  -15,  -15,  -15,   43,  -15,   43,   43,   43,
       43,   46,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       47,   43,   43,  -15,  -15,  -15,  -15,  -15
    },

    {
        3,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,
      -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,

      -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,
      -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,  -16,
      -16,  -16,  -16,  -16,  -16,  -16,   44,  -16,   45,   45,
       45,   45,   45,   45,   45,   45,   45,   45,  -16,  -16,
      -16,  -16,  -16,  -16,  -16,   43,   43,   43,   43,   46,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -16,  -16,  -16,  -16,   43,  -16,   43,   43,   43,
       43,   46,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,  -16,  -16,  -16,  -16,  -16
    },

    {
        3,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,   48,   49,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,

      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17,
      -17,  -17,  -17,  -17,  -17,  -17,  -17,  -17
    },

    {
        3,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,

      -18,   50,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18,
      -18,  -18,  -18,  -18,  -18,  -18,  -18,  -18
    },

    {
        3,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,

      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,   51,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19,
      -19,  -19,  -19,  -19,  -19,  -19,  -19,  -19

    },

    {
        3,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,
      -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,
      -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,
      -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,  -20,
```

```
      -20,  -20,  -20,  -20,  -20,  -20,   43,  -20,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -20,  -20,
      -20,  -20,  -20,  -20,  -20,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   52,   43,
       43,   53,   43,   43,   54,   43,   43,   43,   43,   43,
       43,  -20,  -20,  -20,  -20,   43,  -20,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       52,   43,   43,   53,   43,   43,   54,   43,   43,   43,
       43,   43,   43,  -20,  -20,  -20,  -20,  -20
    },

    {
        3,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,
      -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,
      -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,
      -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,  -21,
      -21,  -21,  -21,  -21,  -21,  -21,   43,  -21,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -21,  -21,
      -21,  -21,  -21,  -21,  -21,   43,   43,   43,   43,   55,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   56,   43,   43,   43,   43,
       43,  -21,  -21,  -21,  -21,   43,  -21,   43,   43,   43,
       43,   55,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   56,   43,   43,
       43,   43,   43,  -21,  -21,  -21,  -21,  -21
    },

    {
        3,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,
      -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,
      -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,
      -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,  -22,

      -22,  -22,  -22,  -22,  -22,  -22,   43,  -22,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -22,  -22,
      -22,  -22,  -22,  -22,  -22,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   57,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -22,  -22,  -22,  -22,   43,  -22,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   57,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -22,  -22,  -22,  -22,  -22
    },

    {
        3,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,

      -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,
      -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,
      -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,  -23,
      -23,  -23,  -23,  -23,  -23,  -23,   43,  -23,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -23,  -23,
      -23,  -23,  -23,  -23,  -23,   43,   43,   43,   43,   43,
       43,   43,   43,   58,   43,   43,   43,   43,   43,   59,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -23,  -23,  -23,  -23,   43,  -23,   43,   43,   43,
       43,   43,   43,   43,   43,   58,   43,   43,   43,   43,

       43,   59,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -23,  -23,  -23,  -23,  -23
    },

    {
        3,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,
      -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,
      -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,
      -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,  -24,
      -24,  -24,  -24,  -24,  -24,  -24,   43,  -24,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -24,  -24,
      -24,  -24,  -24,  -24,  -24,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   60,   43,   61,   43,

       43,   43,   62,   43,   43,   43,   63,   43,   64,   43,
       43,  -24,  -24,  -24,  -24,   43,  -24,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   60,   43,
       61,   43,   43,   43,   62,   43,   43,   43,   63,   43,
       64,   43,   43,  -24,  -24,  -24,  -24,  -24
    },

    {
        3,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,
      -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,
      -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,
      -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,  -25,
      -25,  -25,  -25,  -25,  -25,  -25,   43,  -25,   43,   43,
```

```

      43,   43,   43,   43,   43,   43,   43,   43,  -25,  -25,
     -25,  -25,  -25,  -25,  -25,   65,   43,   43,   43,   43,
      43,   43,   43,   66,   43,   43,   43,   43,   43,   67,
      43,   43,   68,   43,   43,   43,   43,   43,   43,   43,
      43,  -25,  -25,  -25,  -25,   43,  -25,   65,   43,   43,
      43,   43,   43,   43,   43,   66,   43,   43,   43,   43,
      43,   67,   43,   43,   68,   43,   43,   43,   43,   43,
      43,   43,   43,  -25,  -25,  -25,  -25,  -25
    },

    {
       3,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,
     -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,

     -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,
     -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,  -26,
     -26,  -26,  -26,  -26,  -26,  -26,   43,  -26,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -26,  -26,
     -26,  -26,  -26,  -26,  -26,   69,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,  -26,  -26,  -26,  -26,   43,  -26,   69,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,  -26,  -26,  -26,  -26,  -26
    },

    {
       3,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,
     -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,
     -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,
     -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,  -27,
     -27,  -27,  -27,  -27,  -27,  -27,   43,  -27,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -27,  -27,
     -27,  -27,  -27,  -27,  -27,   43,   43,   43,   70,   43,
      71,   43,   43,   43,   43,   43,   43,   43,   72,   43,
      43,   43,   73,   43,   43,   43,   43,   43,   43,   43,

      43,  -27,  -27,  -27,  -27,   43,  -27,   43,   43,   43,
      70,   43,   71,   43,   43,   43,   43,   43,   43,   43,
      72,   43,   43,   43,   73,   43,   43,   43,   43,   43,
      43,   43,   43,  -27,  -27,  -27,  -27,  -27
    },

    {
       3,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,
     -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,
     -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,
     -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,  -28,
     -28,  -28,  -28,  -28,  -28,  -28,   43,  -28,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -28,  -28,

     -28,  -28,  -28,  -28,  -28,   74,   43,   43,   43,   75,
      43,   43,   43,   43,   43,   43,   43,   43,   76,   77,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,  -28,  -28,  -28,  -28,   43,  -28,   74,   43,   43,
      43,   75,   43,   43,   43,   43,   43,   43,   43,   43,
      76,   77,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,  -28,  -28,  -28,  -28,  -28
    },

    {
       3,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,
     -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,
     -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,

     -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,  -29,
     -29,  -29,  -29,  -29,  -29,  -29,   43,  -29,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -29,  -29,
     -29,  -29,  -29,  -29,  -29,   78,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   79,
      43,   43,   80,   43,   43,   43,   43,   43,   43,   43,
      43,  -29,  -29,  -29,  -29,   43,  -29,   78,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   79,   43,   43,   80,   43,   43,   43,   43,   43,
      43,   43,   43,  -29,  -29,  -29,  -29,  -29

    },

    {
       3,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,
     -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,
     -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,
     -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,  -30,
     -30,  -30,  -30,  -30,  -30,  -30,   43,  -30,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,  -30,  -30,
       -30,  -30,  -30,  -30,  -30,   81,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   82,
        43,   83,   84,   43,   43,   85,   43,   43,   43,   43,
        43,  -30,  -30,  -30,  -30,   43,  -30,   81,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   82,   43,   83,   84,   43,   43,   85,   43,   43,
        43,   43,   43,  -30,  -30,  -30,  -30,  -30
    },

    {
         3,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,
       -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,
       -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,
       -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,  -31,
       -31,  -31,  -31,  -31,  -31,  -31,   43,  -31,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -31,  -31,
       -31,  -31,  -31,  -31,  -31,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   86,   43,   43,   43,   43,   43,   43,   43,
        43,  -31,  -31,  -31,  -31,   43,  -31,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   86,   43,   43,   43,   43,   43,
        43,   43,   43,  -31,  -31,  -31,  -31,  -31
    },

    {
         3,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,
       -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,
       -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,
       -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,  -32,

       -32,  -32,  -32,  -32,  -32,  -32,   43,  -32,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -32,  -32,
       -32,  -32,  -32,  -32,  -32,   43,   43,   43,   43,   87,
        43,   43,   43,   43,   43,   43,   43,   43,   88,   89,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -32,  -32,  -32,  -32,   43,  -32,   43,   43,   43,
        43,   87,   43,   43,   43,   43,   43,   43,   43,   43,
        88,   89,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -32,  -32,  -32,  -32,  -32
    },

    {
         3,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,

       -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,
       -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,
       -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,  -33,
       -33,  -33,  -33,  -33,  -33,  -33,   43,  -33,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -33,  -33,
       -33,  -33,  -33,  -33,  -33,   90,   43,   43,   43,   91,
        43,   43,   43,   92,   43,   43,   43,   43,   43,   43,
        43,   93,   43,   43,   94,   43,   43,   43,   43,   43,
        43,  -33,  -33,  -33,  -33,   43,  -33,   90,   43,   43,
        43,   91,   43,   43,   43,   92,   43,   43,   43,   43,

        43,   43,   43,   93,   43,   43,   94,   43,   43,   43,
        43,   43,   43,  -33,  -33,  -33,  -33,  -33
    },

    {
         3,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,
       -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,
       -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,
       -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,  -34,
       -34,  -34,  -34,  -34,  -34,  -34,   43,  -34,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -34,  -34,
       -34,  -34,  -34,  -34,  -34,   95,   43,   43,   43,   43,
        96,   43,   43,   43,   43,   43,   43,   43,   97,   98,

        43,   43,   99,   43,   43,   43,   43,   43,   43,   43,
        43,  -34,  -34,  -34,  -34,   43,  -34,   95,   43,   43,
        43,   43,   96,   43,   43,   43,   43,   43,   43,   43,
        97,   98,   43,   43,   99,   43,   43,   43,   43,   43,
        43,   43,   43,  -34,  -34,  -34,  -34,  -34
    },

    {
         3,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,
       -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,
       -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,
       -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,  -35,
       -35,  -35,  -35,  -35,  -35,  -35,   43,  -35,   43,   43,

```

```
        43,   43,   43,   43,   43,   43,   43,   43,  -35,  -35,
       -35,  -35,  -35,  -35,  -35,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  100,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -35,  -35,  -35,  -35,   43,  -35,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       100,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -35,  -35,  -35,  -35,  -35
    },

    {
         3,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,
       -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,

       -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,
       -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,  -36,
       -36,  -36,  -36,  -36,  -36,  -36,   43,  -36,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -36,  -36,
       -36,  -36,  -36,  -36,  -36,  101,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -36,  -36,  -36,  -36,   43,  -36,  101,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,  -36,  -36,  -36,  -36,  -36
    },

    {
         3,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,
       -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,
       -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,
       -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,  -37,
       -37,  -37,  -37,  -37,  -37,  -37,   43,  -37,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -37,  -37,
       -37,  -37,  -37,  -37,  -37,   43,   43,   43,   43,  102,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,  -37,  -37,  -37,  -37,   43,  -37,   43,   43,   43,
        43,  102,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -37,  -37,  -37,  -37,  -37
    },

    {
         3,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,
       -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,
       -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,
       -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,  -38,
       -38,  -38,  -38,  -38,  -38,  -38,   43,  -38,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -38,  -38,

       -38,  -38,  -38,  -38,  -38,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,  103,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -38,  -38,  -38,  -38,   43,  -38,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  103,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -38,  -38,  -38,  -38,  -38
    },

    {
         3,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,

       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39,
       -39,  -39,  -39,  -39,  -39,  -39,  -39,  -39

    },

    {
         3,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
       -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
       -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
       -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
       -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
       -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
```

```
      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,

      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40,
      -40,  -40,  -40,  -40,  -40,  -40,  -40,  -40
    },

    {
        3,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,

      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41,
      -41,  -41,  -41,  -41,  -41,  -41,  -41,  -41
    },

    {
        3,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,
      -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,
      -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,
      -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,  -42,

      -42,  -42,  -42,  -42,  -42,  -42,   43,  -42,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -42,  -42,
      -42,  -42,  -42,  -42,  -42,   95,   43,   43,   43,   43,
       96,   43,   43,   43,   43,   43,   43,   43,   97,   98,
       43,   43,   99,   43,   43,   43,   43,   43,   43,   43,
       43,  -42,  -42,  -42,  -42,   43,  -42,   95,   43,   43,
       43,   43,   96,   43,   43,   43,   43,   43,   43,   43,
       97,   98,   43,   43,   99,   43,   43,   43,   43,   43,
       43,   43,   43,  -42,  -42,  -42,  -42,  -42
    },

    {
        3,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,

      -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,
      -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,
      -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,  -43,
      -43,  -43,  -43,  -43,  -43,  -43,   43,  -43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -43,  -43,
      -43,  -43,  -43,  -43,  -43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -43,  -43,  -43,  -43,   43,  -43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -43,  -43,  -43,  -43,  -43
    },

    {
        3,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,
      -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,
      -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,
      -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,  -44,
      -44,  -44,  -44,  -44,  -44,  -44,   43,  -44,  104,  104,
      104,  104,  104,  104,  104,  104,  104,  104,  -44,  -44,
      -44,  -44,  -44,  -44,  -44,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -44,  -44,  -44,  -44,   43,  -44,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -44,  -44,  -44,  -44,  -44
    },

    {
        3,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,
      -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,
      -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,
      -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,  -45,
      -45,  -45,  -45,  -45,  -45,  -45,   44,  -45,   45,   45,

       45,   45,   45,   45,   45,   45,   45,   45,  -45,  -45,
```

```
      -45,  -45,  -45,  -45,  -45,   43,   43,   43,   43,   46,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -45,  -45,  -45,  -45,   43,  -45,   43,   43,   43,
       43,   46,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -45,  -45,  -45,  -45,  -45
    },

    {
        3,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,
      -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,

      -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,
      -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,  -46,
      -46,  -46,  -46,  -46,  -46,  105,   43,  -46,  106,  106,
      106,  106,  106,  106,  106,  106,  106,  106,  -46,  -46,
      -46,  -46,  -46,  -46,  -46,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -46,  -46,  -46,  -46,   43,  -46,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,  -46,  -46,  -46,  -46,  -46
    },

    {
        3,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,
      -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,
      -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,
      -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,  -47,
      -47,  -47,  -47,  -47,  -47,  -47,   43,  -47,  107,  107,
      107,  107,  107,  107,  107,  107,  107,  107,  -47,  -47,
      -47,  -47,  -47,  -47,  -47,  107,  107,  107,  107,  107,
      107,  107,  107,  107,  107,  107,  107,  107,  107,  107,
      107,  107,  107,  107,  107,  107,  107,  107,  107,  107,

      107,  108,  108,  108,  108,  107,  108,  107,  107,  107,
      107,  107,  107,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -47,  -47,  -47,  -47,  -47
    },

    {
        3,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,

      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48,
      -48,  -48,  -48,  -48,  -48,  -48,  -48,  -48
    },

    {
        3,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,

      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49,
      -49,  -49,  -49,  -49,  -49,  -49,  -49,  -49

    },

    {
        3,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
```

```
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,

      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50,
      -50,  -50,  -50,  -50,  -50,  -50,  -50,  -50
    },

    {
        3,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,

      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51,
      -51,  -51,  -51,  -51,  -51,  -51,  -51,  -51
    },

    {
        3,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,
      -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,
      -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,
      -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,  -52,

      -52,  -52,  -52,  -52,  -52,  -52,   43,  -52,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -52,  -52,
      -52,  -52,  -52,  -52,  -52,   43,   43,   43,  109,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  -52,  -52,  -52,  -52,   43,  -52,   43,   43,   43,
      109,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  -52,  -52,  -52,  -52,  -52
    },

    {
        3,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,

      -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,
      -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,
      -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,  -53,
      -53,  -53,  -53,  -53,  -53,  -53,   43,  -53,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -53,  -53,
      -53,  -53,  -53,  -53,  -53,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,  110,   43,   43,   43,   43,
       43,  -53,  -53,  -53,  -53,   43,  -53,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,  110,   43,   43,
       43,   43,   43,  -53,  -53,  -53,  -53,  -53
    },

    {
        3,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,
      -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,
      -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,
      -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,  -54,
      -54,  -54,  -54,  -54,  -54,  -54,   43,  -54,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  -54,  -54,
      -54,  -54,  -54,  -54,  -54,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,  111,   43,   43,   43,   43,   43,   43,   43,
       43,  -54,  -54,  -54,  -54,   43,  -54,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  111,   43,   43,   43,   43,   43,
       43,   43,   43,  -54,  -54,  -54,  -54,  -54
    },

    {
        3,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,
      -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,
      -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,
      -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,  -55,
      -55,  -55,  -55,  -55,  -55,  -55,   43,  -55,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,  -55,  -55,
      -55,  -55,  -55,  -55,  -55,   43,   43,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  112,   43,   43,   43,   43,   43,
        43,  -55,  -55,  -55,  -55,   43,  -55,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,  112,   43,   43,   43,
        43,   43,   43,  -55,  -55,  -55,  -55,  -55
    },

    {
         3,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,
       -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,

       -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,
       -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,  -56,
       -56,  -56,  -56,  -56,  -56,  -56,   43,  -56,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -56,  -56,
       -56,  -56,  -56,  -56,  -56,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  113,   43,   43,   43,   43,   43,   43,
        43,  -56,  -56,  -56,  -56,   43,  -56,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  113,   43,   43,   43,   43,

        43,   43,   43,  -56,  -56,  -56,  -56,  -56
    },

    {
         3,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,
       -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,
       -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,
       -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,  -57,
       -57,  -57,  -57,  -57,  -57,  -57,   43,  -57,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -57,  -57,
       -57,  -57,  -57,  -57,  -57,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  114,   43,   43,   43,   43,   43,   43,

        43,  -57,  -57,  -57,  -57,   43,  -57,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  114,   43,   43,   43,   43,
        43,   43,   43,  -57,  -57,  -57,  -57,  -57
    },

    {
         3,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,
       -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,
       -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,
       -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,  -58,
       -58,  -58,  -58,  -58,  -58,  -58,   43,  -58,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -58,  -58,

       -58,  -58,  -58,  -58,  -58,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  115,   43,   43,   43,   43,   43,   43,
        43,  -58,  -58,  -58,  -58,   43,  -58,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  115,   43,   43,   43,   43,
        43,   43,   43,  -58,  -58,  -58,  -58,  -58
    },

    {
         3,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,
       -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,
       -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,

       -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,  -59,
       -59,  -59,  -59,  -59,  -59,  -59,   43,  -59,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -59,  -59,
       -59,  -59,  -59,  -59,  -59,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -59,  -59,  -59,  -59,   43,  -59,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -59,  -59,  -59,  -59,  -59

    },

    {
         3,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,
       -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,
       -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,
       -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,  -60,
       -60,  -60,  -60,  -60,  -60,  -60,   43,  -60,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -60,  -60,
       -60,  -60,  -60,  -60,  -60,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
      43,   43,   43,  116,   43,   43,   43,   43,   43,   43,
      43,  -60,  -60,  -60,  -60,   43,  -60,   43,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,  116,   43,   43,   43,   43,
      43,   43,   43,  -60,  -60,  -60,  -60,  -60
    },

    {
       3,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,
     -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,
     -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,
     -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,  -61,
     -61,  -61,  -61,  -61,  -61,  -61,   43,  -61,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -61,  -61,
     -61,  -61,  -61,  -61,  -61,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,  117,   43,   43,   43,   43,   43,
      43,  -61,  -61,  -61,  -61,   43,  -61,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,  117,   43,   43,   43,
      43,   43,   43,  -61,  -61,  -61,  -61,  -61
    },

    {
       3,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,
     -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,
     -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,
     -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,  -62,

     -62,  -62,  -62,  -62,  -62,  -62,   43,  -62,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -62,  -62,
     -62,  -62,  -62,  -62,  -62,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,  118,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,  -62,  -62,  -62,  -62,   43,  -62,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  118,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,  -62,  -62,  -62,  -62,  -62
    },

    {
       3,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,

     -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,
     -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,
     -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,  -63,
     -63,  -63,  -63,  -63,  -63,  -63,   43,  -63,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -63,  -63,
     -63,  -63,  -63,  -63,  -63,  119,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,  -63,  -63,  -63,  -63,   43,  -63,  119,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,  -63,  -63,  -63,  -63,  -63
    },

    {
       3,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,
     -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,
     -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,
     -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,  -64,
     -64,  -64,  -64,  -64,  -64,  -64,   43,  -64,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  -64,  -64,
     -64,  -64,  -64,  -64,  -64,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

     120,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,  -64,  -64,  -64,  -64,   43,  -64,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,  120,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,  -64,  -64,  -64,  -64,  -64
    },

    {
       3,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,
     -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,
     -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,
     -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,  -65,
     -65,  -65,  -65,  -65,  -65,  -65,   43,  -65,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,  -65,  -65,
     -65,  -65,  -65,  -65,  -65,   43,   43,   43,   43,   43,
      43,   43,   43,  121,   43,   43,  122,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -65,  -65,  -65,  -65,   43,  -65,   43,   43,   43,
        43,   43,   43,   43,   43,  121,   43,   43,  122,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -65,  -65,  -65,  -65,  -65
    },

    {
         3,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,
       -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,

       -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,
       -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,  -66,
       -66,  -66,  -66,  -66,  -66,  -66,   43,  -66,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -66,  -66,
       -66,  -66,  -66,  -66,  -66,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,  123,   43,   43,   43,   43,   43,   43,   43,
        43,  -66,  -66,  -66,  -66,   43,  -66,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  123,   43,   43,   43,   43,   43,

        43,   43,   43,  -66,  -66,  -66,  -66,  -66
    },

    {
         3,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,
       -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,
       -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,
       -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,  -67,
       -67,  -67,  -67,  -67,  -67,  -67,   43,  -67,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -67,  -67,
       -67,  -67,  -67,  -67,  -67,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,  124,   43,   43,   43,   43,   43,   43,   43,

        43,  -67,  -67,  -67,  -67,   43,  -67,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  124,   43,   43,   43,   43,   43,
        43,   43,   43,  -67,  -67,  -67,  -67,  -67
    },

    {
         3,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,
       -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,
       -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,
       -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,  -68,
       -68,  -68,  -68,  -68,  -68,  -68,   43,  -68,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -68,  -68,

       -68,  -68,  -68,  -68,  -68,  125,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -68,  -68,  -68,  -68,   43,  -68,  125,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -68,  -68,  -68,  -68,  -68
    },

    {
         3,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,
       -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,
       -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,

       -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,  -69,
       -69,  -69,  -69,  -69,  -69,  -69,   43,  -69,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -69,  -69,
       -69,  -69,  -69,  -69,  -69,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,  126,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -69,  -69,  -69,  -69,   43,  -69,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,  126,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -69,  -69,  -69,  -69,  -69

    },

    {
         3,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,
       -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,
       -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,
       -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,  -70,
       -70,  -70,  -70,  -70,  -70,  -70,   43,  -70,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -70,  -70,
       -70,  -70,  -70,  -70,  -70,   43,   43,   43,   43,  127,
        43,   43,   43,   43,   43,   43,  128,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
        43,  -70,  -70,  -70,  -70,   43,  -70,   43,   43,   43,

        43,  127,   43,   43,   43,   43,   43,   43,  128,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -70,  -70,  -70,  -70,  -70
    },

    {
         3,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,
      -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,
      -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,
      -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,  -71,
      -71,  -71,  -71,  -71,  -71,  -71,   43,  -71,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -71,  -71,
      -71,  -71,  -71,  -71,  -71,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -71,  -71,  -71,  -71,   43,  -71,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -71,  -71,  -71,  -71,  -71
    },

    {
         3,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,
      -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,
      -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,
      -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,  -72,

      -72,  -72,  -72,  -72,  -72,  -72,   43,  -72,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -72,  -72,
      -72,  -72,  -72,  -72,  -72,  129,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -72,  -72,  -72,  -72,   43,  -72,  129,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -72,  -72,  -72,  -72,  -72
    },

    {
         3,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,

      -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,
      -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,
      -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,  -73,
      -73,  -73,  -73,  -73,  -73,  -73,   43,  -73,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -73,  -73,
      -73,  -73,  -73,  -73,  -73,   43,   43,   43,   43,   43,
      130,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -73,  -73,  -73,  -73,   43,  -73,   43,   43,   43,
        43,   43,  130,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -73,  -73,  -73,  -73,  -73
    },

    {
         3,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,
      -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,
      -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,
      -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,  -74,
      -74,  -74,  -74,  -74,  -74,  -74,   43,  -74,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -74,  -74,
      -74,  -74,  -74,  -74,  -74,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,  131,   43,   43,   43,   43,   43,   43,
        43,  -74,  -74,  -74,  -74,   43,  -74,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  131,   43,   43,   43,   43,
        43,   43,   43,  -74,  -74,  -74,  -74,  -74
    },

    {
         3,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,
      -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,
      -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,
      -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,  -75,
      -75,  -75,  -75,  -75,  -75,  -75,   43,  -75,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,  -75,  -75,
      -75,  -75,  -75,  -75,  -75,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  132,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
        43,  -75,  -75,  -75,  -75,   43,  -75,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       132,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -75,  -75,  -75,  -75,  -75
    },

    {
         3,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,
       -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,

       -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,
       -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,  -76,
       -76,  -76,  -76,  -76,  -76,  -76,   43,  -76,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -76,  -76,
       -76,  -76,  -76,  -76,  -76,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -76,  -76,  -76,  -76,   43,  -76,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,  -76,  -76,  -76,  -76,  -76
    },

    {
         3,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,
       -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,
       -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,
       -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,  -77,
       -77,  -77,  -77,  -77,  -77,  -77,   43,  -77,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -77,  -77,
       -77,  -77,  -77,  -77,  -77,   43,   43,   43,   43,   43,
        43,  133,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,  -77,  -77,  -77,  -77,   43,  -77,   43,   43,   43,
        43,   43,   43,  133,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -77,  -77,  -77,  -77,  -77
    },

    {
         3,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,
       -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,
       -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,
       -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,  -78,
       -78,  -78,  -78,  -78,  -78,  -78,   43,  -78,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -78,  -78,

       -78,  -78,  -78,  -78,  -78,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  134,   43,
        43,  -78,  -78,  -78,  -78,   43,  -78,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       134,   43,   43,  -78,  -78,  -78,  -78,  -78
    },

    {
         3,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,
       -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,
       -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,

       -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,  -79,
       -79,  -79,  -79,  -79,  -79,  -79,   43,  -79,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -79,  -79,
       -79,  -79,  -79,  -79,  -79,   43,   43,   43,  135,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -79,  -79,  -79,  -79,   43,  -79,   43,   43,   43,
       135,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -79,  -79,  -79,  -79,  -79

    },

    {
         3,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,
       -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,
       -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,
       -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,  -80,
       -80,  -80,  -80,  -80,  -80,  -80,   43,  -80,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -80,  -80,
       -80,  -80,  -80,  -80,  -80,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -80,  -80,  -80,  -80,   43,  -80,   43,   43,   43,
```

```
01741
01742        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01743        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01744        43,   43,   43,  -80,  -80,  -80,  -80,  -80
01745     },
01746
01747     {
01748         3,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,
01749       -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,
01750       -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,
01751       -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,  -81,
01752       -81,  -81,  -81,  -81,  -81,  -81,   43,  -81,   43,   43,
01753        43,   43,   43,   43,   43,   43,   43,   43,  -81,  -81,
01754       -81,  -81,  -81,  -81,  -81,   43,   43,   43,   43,   43,
01755
01756        43,   43,   43,   43,   43,   43,   43,   43,  136,   43,
01757        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01758        43,  -81,  -81,  -81,  -81,   43,  -81,   43,   43,   43,
01759        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01760       136,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01761        43,   43,   43,  -81,  -81,  -81,  -81,  -81
01762     },
01763
01764     {
01765         3,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,
01766       -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,
01767       -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,
01768       -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,  -82,
01769
01770       -82,  -82,  -82,  -82,  -82,  -82,   43,  -82,   43,   43,
01771        43,   43,   43,   43,   43,   43,   43,   43,  -82,  -82,
01772       -82,  -82,  -82,  -82,  -82,   43,   43,   43,   43,   43,
01773        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01774        43,   43,  137,   43,  138,   43,   43,   43,   43,   43,
01775        43,  -82,  -82,  -82,  -82,   43,  -82,   43,   43,   43,
01776        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01777        43,   43,   43,   43,  137,   43,  138,   43,   43,   43,
01778        43,   43,   43,  -82,  -82,  -82,  -82,  -82
01779     },
01780
01781     {
01782         3,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,
01783
01784       -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,
01785       -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,
01786       -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,  -83,
01787       -83,  -83,  -83,  -83,  -83,  -83,   43,  -83,   43,   43,
01788        43,   43,   43,   43,   43,   43,   43,   43,  -83,  -83,
01789       -83,  -83,  -83,  -83,  -83,   43,   43,   43,   43,   43,
01790        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01791        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01792        43,  -83,  -83,  -83,  -83,   43,  -83,   43,   43,   43,
01793        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01794
01795        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01796        43,   43,   43,  -83,  -83,  -83,  -83,  -83
01797     },
01798
01799     {
01800         3,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,
01801       -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,
01802       -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,
01803       -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,  -84,
01804       -84,  -84,  -84,  -84,  -84,  -84,   43,  -84,   43,   43,
01805        43,   43,   43,   43,   43,   43,   43,   43,  -84,  -84,
01806       -84,  -84,  -84,  -84,  -84,   43,   43,   43,   43,   43,
01807        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01808
01809        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01810        43,  -84,  -84,  -84,  -84,   43,  -84,   43,   43,   43,
01811        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01812        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01813        43,   43,   43,  -84,  -84,  -84,  -84,  -84
01814     },
01815
01816     {
01817         3,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,
01818       -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,
01819       -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,
01820       -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,  -85,
01821       -85,  -85,  -85,  -85,  -85,  -85,   43,  -85,   43,   43,
01822
01823        43,   43,   43,   43,   43,   43,   43,   43,  -85,  -85,
01824       -85,  -85,  -85,  -85,  -85,   43,   43,   43,   43,   43,
01825        43,   43,   43,   43,   43,   43,   43,  139,   43,   43,
01826        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
01827        43,  -85,  -85,  -85,  -85,   43,  -85,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,  139,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -85,  -85,  -85,  -85  -85
    },

    {
         3,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,
       -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,

       -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,
       -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,  -86,
       -86,  -86,  -86,  -86,  -86,  -86,   43,  -86,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -86,  -86,
       -86,  -86,  -86,  -86,  -86,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -86,  -86,  -86,  -86,   43,  -86,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,  -86,  -86,  -86,  -86,  -86
    },

    {
         3,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,
       -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,
       -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,
       -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,  -87,
       -87,  -87,  -87,  -87,  -87,  -87,   43,  -87,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -87,  -87,
       -87,  -87,  -87,  -87,  -87,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  140,   43,   43,   43,   43,   43,   43,

        43,  -87,  -87,  -87,  -87,   43,  -87,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  140,   43,   43,   43,   43,
        43,   43,   43,  -87,  -87,  -87,  -87,  -87
    },

    {
         3,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,
       -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,
       -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,
       -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,  -88,
       -88,  -88,  -88,  -88,  -88,  -88,   43,  -88,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -88,  -88,

       -88,  -88,  -88,  -88,  -88,   43,   43,   43,  141,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -88,  -88,  -88,  -88,   43,  -88,   43,   43,   43,
       141,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -88,  -88,  -88,  -88,  -88
    },

    {
         3,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,
       -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,
       -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,

       -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,  -89,
       -89,  -89,  -89,  -89,  -89,  -89,   43,  -89,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -89,  -89,
       -89,  -89,  -89,  -89,  -89,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  142,   43,   43,   43,   43,
        43,  -89,  -89,  -89,  -89,   43,  -89,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,  142,   43,   43,
        43,   43,   43,  -89,  -89,  -89,  -89,  -89

    },

    {
         3,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,
       -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,
       -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,
       -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,  -90,
       -90,  -90,  -90,  -90,  -90,  -90,   43,  -90,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -90,  -90,
       -90,  -90,  -90,  -90,  -90,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  143,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -90,  -90,  -90,  -90,   43,  -90,   43,   43,   43,

```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  143,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -90,  -90,  -90,  -90,  -90
    },

    {
         3,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,
       -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,
       -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,
       -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,  -91,
       -91,  -91,  -91,  -91,  -91,  -91,   43,  -91,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -91,  -91,
       -91,  -91,  -91,  -91,  -91,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  144,   43,   43,   43,   43,   43,
        43,  -91,  -91,  -91,  -91,   43,  -91,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,  144,   43,   43,   43,
        43,   43,   43,  -91,  -91,  -91,  -91,  -91
    },

    {
         3,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,
       -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,
       -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,
       -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,  -92,

       -92,  -92,  -92,  -92,  -92,  -92,   43,  -92,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -92,  -92,
       -92,  -92,  -92,  -92,  -92,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  145,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -92,  -92,  -92,  -92,   43,  -92,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       145,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -92,  -92,  -92,  -92,  -92
    },

    {
         3,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,

       -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,
       -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,
       -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,  -93,
       -93,  -93,  -93,  -93,  -93,  -93,   43,  -93,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -93,  -93,
       -93,  -93,  -93,  -93,  -93,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,  146,   43,   43,   43,   43,   43,   43,   43,
        43,  -93,  -93,  -93,  -93,   43,  -93,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,  146,   43,   43,   43,   43,   43,
        43,   43,   43,  -93,  -93,  -93,  -93,  -93
    },

    {
         3,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,
       -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,
       -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,
       -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,  -94,
       -94,  -94,  -94,  -94,  -94,  -94,   43,  -94,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -94,  -94,
       -94,  -94,  -94,  -94,  -94,  147,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -94,  -94,  -94,  -94,   43,  -94,  147,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -94,  -94,  -94,  -94,  -94
    },

    {
         3,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,
       -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,
       -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,
       -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,  -95,
       -95,  -95,  -95,  -95,  -95,  -95,   43,  -95,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,  -95,  -95,
       -95,  -95,  -95,  -95,  -95,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,  148,   43,   43,   43,
        43,  -95,  -95,  -95,  -95,   43,  -95,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,  148,   43,
        43,   43,   43,  -95,  -95,  -95,  -95,  -95
    },

    {
         3,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,
       -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,

       -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,
       -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,  -96,
       -96,  -96,  -96,  -96,  -96,  -96,   43,  -96,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -96,  -96,
       -96,  -96,  -96,  -96,  -96,   43,   43,   43,   43,   43,
        43,   43,   43,  149,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -96,  -96,  -96,  -96,   43,  -96,   43,   43,   43,
        43,   43,   43,   43,   43,  149,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,  -96,  -96,  -96,  -96,  -96
    },

    {
         3,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,
       -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,
       -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,
       -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,  -97,
       -97,  -97,  -97,  -97,  -97,  -97,   43,  -97,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -97,  -97,
       -97,  -97,  -97,  -97,  -97,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,  150,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,  -97,  -97,  -97,  -97,   43,  -97,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  150,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -97,  -97,  -97,  -97,  -97
    },

    {
         3,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,
       -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,
       -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,
       -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,  -98,
       -98,  -98,  -98,  -98,  -98,  -98,   43,  -98,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -98,  -98,

       -98,  -98,  -98,  -98,  -98,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  -98,  -98,  -98,  -98,   43,  -98,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  -98,  -98,  -98,  -98,  -98
    },

    {
         3,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,
       -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,
       -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,

       -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,  -99,
       -99,  -99,  -99,  -99,  -99,  -99,   43,  -99,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  -99,  -99,
       -99,  -99,  -99,  -99,  -99,   43,   43,   43,   43,   43,
        43,   43,   43,  151,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  152,   43,   43,   43,   43,
        43,  -99,  -99,  -99,  -99,   43,  -99,   43,   43,   43,
        43,   43,   43,   43,   43,  151,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,  152,   43,   43,
        43,   43,   43,  -99,  -99,  -99,  -99,  -99

    },

    {
         3, -100, -100, -100, -100, -100, -100, -100, -100, -100,
      -100, -100, -100, -100, -100, -100, -100, -100, -100, -100,
      -100, -100, -100, -100, -100, -100, -100, -100, -100, -100,
      -100, -100, -100, -100, -100, -100, -100, -100, -100, -100,
      -100, -100, -100, -100, -100, -100,   43, -100,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -100, -100,
      -100, -100, -100, -100, -100,   43,   43,   43,   43,   43,
        43,   43,   43,  153,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -100, -100, -100, -100,   43, -100,   43,   43,   43,

        43,   43,   43,   43,   43,  153,   43,   43,   43,   43,
```

```
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -100, -100, -100, -100, -100
    },

    {
       3, -101, -101, -101, -101, -101, -101, -101, -101, -101,
    -101, -101, -101, -101, -101, -101, -101, -101, -101, -101,
    -101, -101, -101, -101, -101, -101, -101, -101, -101, -101,
    -101, -101, -101, -101, -101, -101, -101, -101, -101, -101,
    -101, -101, -101, -101, -101, -101,   43, -101,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -101, -101,
    -101, -101, -101, -101, -101,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,   43,  154,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -101, -101, -101, -101,   43, -101,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,  154,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -101, -101, -101, -101, -101
    },

    {
       3, -102, -102, -102, -102, -102, -102, -102, -102, -102,
    -102, -102, -102, -102, -102, -102, -102, -102, -102, -102,
    -102, -102, -102, -102, -102, -102, -102, -102, -102, -102,
    -102, -102, -102, -102, -102, -102, -102, -102, -102, -102,

    -102, -102, -102, -102, -102, -102,   43, -102,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -102, -102,
    -102, -102, -102, -102, -102,   43,   43,   43,   43,   43,
      43,   43,   43,  155,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -102, -102, -102, -102,   43, -102,   43,   43,   43,
      43,   43,   43,   43,   43,  155,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -102, -102, -102, -102, -102
    },

    {
       3, -103, -103, -103, -103, -103, -103, -103, -103, -103,

    -103, -103, -103, -103, -103, -103, -103, -103, -103, -103,
    -103, -103, -103, -103, -103, -103, -103, -103, -103, -103,
    -103, -103, -103, -103, -103, -103, -103, -103, -103, -103,
    -103, -103, -103, -103, -103, -103,   43, -103,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -103, -103,
    -103, -103, -103, -103, -103,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,  156,   43,   43,   43,   43,   43,   43,   43,
      43, -103, -103, -103, -103,   43, -103,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,   43,  156,   43,   43,   43,   43,   43,
      43,   43,   43, -103, -103, -103, -103, -103
    },

    {
       3, -104, -104, -104, -104, -104, -104, -104, -104, -104,
    -104, -104, -104, -104, -104, -104, -104, -104, -104, -104,
    -104, -104, -104, -104, -104, -104, -104, -104, -104, -104,
    -104, -104, -104, -104, -104, -104, -104, -104, -104, -104,
    -104, -104, -104, -104, -104, -104,   43, -104,  104,  104,
     104,  104,  104,  104,  104,  104,  104,  104, -104, -104,
    -104, -104, -104, -104, -104,   43,   43,   43,   43,  157,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -104, -104, -104, -104,   43, -104,   43,   43,   43,
      43,  157,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -104, -104, -104, -104, -104
    },

    {
       3, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105,  158,  158,

     158,  158,  158,  158,  158,  158,  158,  158, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
    -105, -105, -105, -105, -105, -105, -105, -105, -105, -105,
```

```
 -105, -105, -105, -105, -105, -105, -105, -105
},

{
    3, -106, -106, -106, -106, -106, -106, -106, -106, -106,
 -106, -106, -106, -106, -106, -106, -106, -106, -106, -106,

 -106, -106, -106, -106, -106, -106, -106, -106, -106, -106,
 -106, -106, -106, -106, -106, -106, -106, -106, -106, -106,
 -106, -106, -106, -106, -106, -106,   43, -106,  106,  106,
  106,  106,  106,  106,  106,  106,  106,  106, -106, -106,
 -106, -106, -106, -106, -106,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -106, -106, -106, -106,   43, -106,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43, -106, -106, -106, -106, -106
},

{
    3, -107, -107, -107, -107, -107, -107, -107, -107, -107,
 -107, -107, -107, -107, -107, -107, -107, -107, -107, -107,
 -107, -107, -107, -107, -107, -107, -107, -107, -107, -107,
 -107, -107, -107, -107, -107, -107, -107, -107, -107, -107,
 -107, -107, -107, -107, -107, -107,   43, -107,  107,  107,
  107,  107,  107,  107,  107,  107,  107,  107, -107, -107,
 -107, -107, -107, -107, -107,  107,  107,  107,  107,  107,
  107,  107,  107,  107,  107,  107,  107,  107,  107,  107,
  107,  107,  107,  107,  107,  107,  107,  107,  107,  107,

  107,  108,  108,  108,  108,  107,  108,  107,  107,  107,
  107,  107,  107,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -107, -107, -107, -107, -107
},

{
    3, -108, -108, -108, -108, -108, -108, -108, -108, -108,
 -108, -108, -108, -108, -108, -108, -108, -108, -108, -108,
 -108, -108, -108, -108, -108, -108, -108, -108, -108, -108,
 -108, -108, -108, -108, -108, -108, -108, -108, -108, -108,
 -108, -108, -108, -108, -108, -108, -108, -108,  108,  108,
  108,  108,  108,  108,  108,  108,  108,  108, -108, -108,

 -108, -108, -108, -108, -108,  108,  108,  108,  108,  108,
  108,  108,  108,  108,  108,  108,  108,  108,  108,  108,
  108,  108,  108,  108,  108,  108,  108,  108,  108,  108,
  108,  108,  108,  108,  108,  108,  108,  108,  108,  108,
  108,  108,  108, -108, -108, -108, -108, -108, -108, -108,
 -108, -108, -108, -108, -108, -108, -108, -108, -108, -108,
 -108, -108, -108, -108, -108, -108, -108, -108
},

{
    3, -109, -109, -109, -109, -109, -109, -109, -109, -109,
 -109, -109, -109, -109, -109, -109, -109, -109, -109, -109,
 -109, -109, -109, -109, -109, -109, -109, -109, -109, -109,

 -109, -109, -109, -109, -109, -109, -109, -109, -109, -109,
 -109, -109, -109, -109, -109, -109,   43, -109,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -109, -109,
 -109, -109, -109, -109, -109,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -109, -109, -109, -109,   43, -109,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -109, -109, -109, -109, -109

},

{
    3, -110, -110, -110, -110, -110, -110, -110, -110, -110,
 -110, -110, -110, -110, -110, -110, -110, -110, -110, -110,
 -110, -110, -110, -110, -110, -110, -110, -110, -110, -110,
 -110, -110, -110, -110, -110, -110, -110, -110, -110, -110,
 -110, -110, -110, -110, -110, -110,   43, -110,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -110, -110,
 -110, -110, -110, -110, -110,   43,   43,   43,   43,  159,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -110, -110, -110, -110,   43, -110,   43,   43,   43,

   43,  159,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
        43,   43,   43, -110, -110, -110, -110, -110
    },

    {
         3, -111, -111, -111, -111, -111, -111, -111, -111, -111,
     -111, -111, -111, -111, -111, -111, -111, -111, -111, -111,
     -111, -111, -111, -111, -111, -111, -111, -111, -111, -111,
     -111, -111, -111, -111, -111, -111, -111, -111, -111, -111,
     -111, -111, -111, -111, -111, -111,   43, -111,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -111, -111,
     -111, -111, -111, -111, -111,   43,   43,   43,   43,   43,

        43,  160,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -111, -111, -111, -111,   43, -111,   43,   43,   43,
        43,   43,   43,  160,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -111, -111, -111, -111, -111
    },

    {
         3, -112, -112, -112, -112, -112, -112, -112, -112, -112,
     -112, -112, -112, -112, -112, -112, -112, -112, -112, -112,
     -112, -112, -112, -112, -112, -112, -112, -112, -112, -112,
     -112, -112, -112, -112, -112, -112, -112, -112, -112, -112,

     -112, -112, -112, -112, -112, -112,   43, -112,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -112, -112,
     -112, -112, -112, -112, -112,  161,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -112, -112, -112, -112,   43, -112,  161,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -112, -112, -112, -112, -112
    },

    {
         3, -113, -113, -113, -113, -113, -113, -113, -113, -113,

     -113, -113, -113, -113, -113, -113, -113, -113, -113, -113,
     -113, -113, -113, -113, -113, -113, -113, -113, -113, -113,
     -113, -113, -113, -113, -113, -113, -113, -113, -113, -113,
     -113, -113, -113, -113, -113, -113,   43, -113,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -113, -113,
     -113, -113, -113, -113, -113,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,  162,
        43, -113, -113, -113, -113,   43, -113,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  162,   43, -113, -113, -113, -113, -113
    },

    {
         3, -114, -114, -114, -114, -114, -114, -114, -114, -114,
     -114, -114, -114, -114, -114, -114, -114, -114, -114, -114,
     -114, -114, -114, -114, -114, -114, -114, -114, -114, -114,
     -114, -114, -114, -114, -114, -114, -114, -114, -114, -114,
     -114, -114, -114, -114, -114, -114,   43, -114,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -114, -114,
     -114, -114, -114, -114, -114,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -114, -114, -114, -114,   43, -114,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -114, -114, -114, -114, -114
    },

    {
         3, -115, -115, -115, -115, -115, -115, -115, -115, -115,
     -115, -115, -115, -115, -115, -115, -115, -115, -115, -115,
     -115, -115, -115, -115, -115, -115, -115, -115, -115, -115,
     -115, -115, -115, -115, -115, -115, -115, -115, -115, -115,
     -115, -115, -115, -115, -115, -115,   43, -115,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43, -115, -115,
     -115, -115, -115, -115, -115,   43,   43,  163,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -115, -115, -115, -115,   43, -115,   43,   43,  163,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -115, -115, -115, -115, -115
```

```
    },

    {
        3, -116, -116, -116, -116, -116, -116, -116, -116, -116,
     -116, -116, -116, -116, -116, -116, -116, -116, -116, -116,

     -116, -116, -116, -116, -116, -116, -116, -116, -116, -116,
     -116, -116, -116, -116, -116, -116, -116, -116, -116, -116,
     -116, -116, -116, -116, -116, -116,   43, -116,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -116, -116,
     -116, -116, -116, -116, -116,   43,   43,   43,   43,  164,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -116, -116, -116, -116,   43, -116,   43,   43,   43,
       43,  164,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -116, -116, -116, -116, -116
    },

    {
        3, -117, -117, -117, -117, -117, -117, -117, -117, -117,
     -117, -117, -117, -117, -117, -117, -117, -117, -117, -117,
     -117, -117, -117, -117, -117, -117, -117, -117, -117, -117,
     -117, -117, -117, -117, -117, -117, -117, -117, -117, -117,
     -117, -117, -117, -117, -117, -117,   43, -117,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -117, -117,
     -117, -117, -117, -117, -117,  165,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -117, -117, -117, -117,   43, -117,  165,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -117, -117, -117, -117, -117
    },

    {
        3, -118, -118, -118, -118, -118, -118, -118, -118, -118,
     -118, -118, -118, -118, -118, -118, -118, -118, -118, -118,
     -118, -118, -118, -118, -118, -118, -118, -118, -118, -118,
     -118, -118, -118, -118, -118, -118, -118, -118, -118, -118,
     -118, -118, -118, -118, -118, -118,   43, -118,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -118, -118,

     -118, -118, -118, -118, -118,  166,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -118, -118, -118, -118,   43, -118,  166,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -118, -118, -118, -118, -118
    },

    {
        3, -119, -119, -119, -119, -119, -119, -119, -119, -119,
     -119, -119, -119, -119, -119, -119, -119, -119, -119, -119,
     -119, -119, -119, -119, -119, -119, -119, -119, -119, -119,

     -119, -119, -119, -119, -119, -119, -119, -119, -119, -119,
     -119, -119, -119, -119, -119, -119,   43, -119,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -119, -119,
     -119, -119, -119, -119, -119,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  167,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -119, -119, -119, -119,   43, -119,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  167,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -119, -119, -119, -119, -119

    },

    {
        3, -120, -120, -120, -120, -120, -120, -120, -120, -120,
     -120, -120, -120, -120, -120, -120, -120, -120, -120, -120,
     -120, -120, -120, -120, -120, -120, -120, -120, -120, -120,
     -120, -120, -120, -120, -120, -120, -120, -120, -120, -120,
     -120, -120, -120, -120, -120, -120,   43, -120,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -120, -120,
     -120, -120, -120, -120, -120,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,  168,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -120, -120, -120, -120,   43, -120,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,  168,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -120, -120, -120, -120, -120
```

```
    },

    {
        3, -121, -121, -121, -121, -121, -121, -121, -121, -121,
     -121, -121, -121, -121, -121, -121, -121, -121, -121, -121,
     -121, -121, -121, -121, -121, -121, -121, -121, -121, -121,
     -121, -121, -121, -121, -121, -121, -121, -121, -121, -121,
     -121, -121, -121, -121, -121, -121,   43, -121,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -121, -121,
     -121, -121, -121, -121, -121,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,  169,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -121, -121, -121, -121,   43, -121,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  169,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -121, -121, -121, -121, -121
    },

    {
        3, -122, -122, -122, -122, -122, -122, -122, -122, -122,
     -122, -122, -122, -122, -122, -122, -122, -122, -122, -122,
     -122, -122, -122, -122, -122, -122, -122, -122, -122, -122,
     -122, -122, -122, -122, -122, -122, -122, -122, -122, -122,

     -122, -122, -122, -122, -122, -122,   43, -122,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -122, -122,
     -122, -122, -122, -122, -122,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  170,   43,   43,   43,   43,   43,   43,
       43, -122, -122, -122, -122,   43, -122,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,  170,   43,   43,   43,   43,
       43,   43,   43, -122, -122, -122, -122, -122
    },

    {
        3, -123, -123, -123, -123, -123, -123, -123, -123, -123,

     -123, -123, -123, -123, -123, -123, -123, -123, -123, -123,
     -123, -123, -123, -123, -123, -123, -123, -123, -123, -123,
     -123, -123, -123, -123, -123, -123, -123, -123, -123, -123,
     -123, -123, -123, -123, -123, -123,   43, -123,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -123, -123,
     -123, -123, -123, -123, -123,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  171,   43,   43,   43,   43,   43,   43,
       43, -123, -123, -123, -123,   43, -123,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,  171,   43,   43,   43,   43,
       43,   43,   43, -123, -123, -123, -123, -123
    },

    {
        3, -124, -124, -124, -124, -124, -124, -124, -124, -124,
     -124, -124, -124, -124, -124, -124, -124, -124, -124, -124,
     -124, -124, -124, -124, -124, -124, -124, -124, -124, -124,
     -124, -124, -124, -124, -124, -124, -124, -124, -124, -124,
     -124, -124, -124, -124, -124, -124,   43, -124,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -124, -124,
     -124, -124, -124, -124, -124,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -124, -124, -124, -124,   43, -124,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -124, -124, -124, -124, -124
    },

    {
        3, -125, -125, -125, -125, -125, -125, -125, -125, -125,
     -125, -125, -125, -125, -125, -125, -125, -125, -125, -125,
     -125, -125, -125, -125, -125, -125, -125, -125, -125, -125,
     -125, -125, -125, -125, -125, -125, -125, -125, -125, -125,
     -125, -125, -125, -125, -125, -125,   43, -125,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43, -125, -125,
     -125, -125, -125, -125, -125,   43,   43,  172,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -125, -125, -125, -125,   43, -125,   43,   43,  172,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -125, -125, -125, -125, -125
    },
```

```
    {
        3, -126, -126, -126, -126, -126, -126, -126, -126, -126,
     -126, -126, -126, -126, -126, -126, -126, -126, -126, -126,

     -126, -126, -126, -126, -126, -126, -126, -126, -126, -126,
     -126, -126, -126, -126, -126, -126, -126, -126, -126, -126,
     -126, -126, -126, -126, -126, -126,   43, -126,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -126, -126,
     -126, -126, -126, -126, -126,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,  173,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -126, -126, -126, -126,   43, -126,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,  173,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -126, -126, -126, -126, -126
    },

    {
        3, -127, -127, -127, -127, -127, -127, -127, -127, -127,
     -127, -127, -127, -127, -127, -127, -127, -127, -127, -127,
     -127, -127, -127, -127, -127, -127, -127, -127, -127, -127,
     -127, -127, -127, -127, -127, -127, -127, -127, -127, -127,
     -127, -127, -127, -127, -127, -127,   43, -127,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -127, -127,
     -127, -127, -127, -127, -127,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  174,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -127, -127, -127, -127,   43, -127,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      174,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -127, -127, -127, -127, -127
    },

    {
        3, -128, -128, -128, -128, -128, -128, -128, -128, -128,
     -128, -128, -128, -128, -128, -128, -128, -128, -128, -128,
     -128, -128, -128, -128, -128, -128, -128, -128, -128, -128,
     -128, -128, -128, -128, -128, -128, -128, -128, -128, -128,
     -128, -128, -128, -128, -128, -128,   43, -128,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -128, -128,

     -128, -128, -128, -128, -128,   43,   43,   43,   43,  175,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -128, -128, -128, -128,   43, -128,   43,   43,   43,
       43,  175,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -128, -128, -128, -128, -128
    },

    {
        3, -129, -129, -129, -129, -129, -129, -129, -129, -129,
     -129, -129, -129, -129, -129, -129, -129, -129, -129, -129,
     -129, -129, -129, -129, -129, -129, -129, -129, -129, -129,

     -129, -129, -129, -129, -129, -129, -129, -129, -129, -129,
     -129, -129, -129, -129, -129, -129,   43, -129,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -129, -129,
     -129, -129, -129, -129, -129,   43,   43,  176,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -129, -129, -129, -129,   43, -129,   43,   43,  176,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -129, -129, -129, -129, -129

    },

    {
        3, -130, -130, -130, -130, -130, -130, -130, -130, -130,
     -130, -130, -130, -130, -130, -130, -130, -130, -130, -130,
     -130, -130, -130, -130, -130, -130, -130, -130, -130, -130,
     -130, -130, -130, -130, -130, -130, -130, -130, -130, -130,
     -130, -130, -130, -130, -130, -130,   43, -130,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -130, -130,
     -130, -130, -130, -130, -130,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -130, -130, -130, -130,   43, -130,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -130, -130, -130, -130, -130
    },
```

```

    {
        3, -131, -131, -131, -131, -131, -131, -131, -131, -131,
     -131, -131, -131, -131, -131, -131, -131, -131, -131, -131,
     -131, -131, -131, -131, -131, -131, -131, -131, -131, -131,
     -131, -131, -131, -131, -131, -131, -131, -131, -131, -131,
     -131, -131, -131, -131, -131, -131,   43, -131,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -131, -131,
     -131, -131, -131, -131, -131,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  177,   43,   43,   43,   43,   43,
       43, -131, -131, -131, -131,   43, -131,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  177,   43,   43,   43,
       43,   43,   43, -131, -131, -131, -131, -131
    },

    {
        3, -132, -132, -132, -132, -132, -132, -132, -132, -132,
     -132, -132, -132, -132, -132, -132, -132, -132, -132, -132,
     -132, -132, -132, -132, -132, -132, -132, -132, -132, -132,
     -132, -132, -132, -132, -132, -132, -132, -132, -132, -132,

     -132, -132, -132, -132, -132, -132,   43, -132,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -132, -132,
     -132, -132, -132, -132, -132,   43,   43,   43,   43,   43,
       43,  178,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -132, -132, -132, -132,   43, -132,   43,   43,   43,
       43,   43,   43,  178,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -132, -132, -132, -132, -132
    },

    {
        3, -133, -133, -133, -133, -133, -133, -133, -133, -133,

     -133, -133, -133, -133, -133, -133, -133, -133, -133, -133,
     -133, -133, -133, -133, -133, -133, -133, -133, -133, -133,
     -133, -133, -133, -133, -133, -133, -133, -133, -133, -133,
     -133, -133, -133, -133, -133, -133,   43, -133,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -133, -133,
     -133, -133, -133, -133, -133,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,  179,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -133, -133, -133, -133,   43, -133,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      179,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -133, -133, -133, -133, -133
    },

    {
        3, -134, -134, -134, -134, -134, -134, -134, -134, -134,
     -134, -134, -134, -134, -134, -134, -134, -134, -134, -134,
     -134, -134, -134, -134, -134, -134, -134, -134, -134, -134,
     -134, -134, -134, -134, -134, -134, -134, -134, -134, -134,
     -134, -134, -134, -134, -134, -134,   43, -134,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -134, -134,
     -134, -134, -134, -134, -134,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,  180,   43,   43,   43,   43,   43,   43,   43,
       43, -134, -134, -134, -134,   43, -134,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  180,   43,   43,   43,   43,   43,
       43,   43,   43, -134, -134, -134, -134, -134
    },

    {
        3, -135, -135, -135, -135, -135, -135, -135, -135, -135,
     -135, -135, -135, -135, -135, -135, -135, -135, -135, -135,
     -135, -135, -135, -135, -135, -135, -135, -135, -135, -135,
     -135, -135, -135, -135, -135, -135, -135, -135, -135, -135,
     -135, -135, -135, -135, -135, -135,   43, -135,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43, -135, -135,
     -135, -135, -135, -135, -135,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -135, -135, -135, -135,   43, -135,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -135, -135, -135, -135, -135
    },

```

```
    {
        3, -136, -136, -136, -136, -136, -136, -136, -136, -136,
     -136, -136, -136, -136, -136, -136, -136, -136, -136, -136,

     -136, -136, -136, -136, -136, -136, -136, -136, -136, -136,
     -136, -136, -136, -136, -136, -136, -136, -136, -136, -136,
     -136, -136, -136, -136, -136, -136,   43, -136,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -136, -136,
     -136, -136, -136, -136, -136,   43,   43,   43,  181,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -136, -136, -136, -136,   43, -136,   43,   43,   43,
      181,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -136, -136, -136, -136, -136
    },

    {
        3, -137, -137, -137, -137, -137, -137, -137, -137, -137,
     -137, -137, -137, -137, -137, -137, -137, -137, -137, -137,
     -137, -137, -137, -137, -137, -137, -137, -137, -137, -137,
     -137, -137, -137, -137, -137, -137, -137, -137, -137, -137,
     -137, -137, -137, -137, -137, -137,   43, -137,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -137, -137,
     -137, -137, -137, -137, -137,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,  182,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -137, -137, -137, -137,   43, -137,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,  182,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -137, -137, -137, -137, -137
    },

    {
        3, -138, -138, -138, -138, -138, -138, -138, -138, -138,
     -138, -138, -138, -138, -138, -138, -138, -138, -138, -138,
     -138, -138, -138, -138, -138, -138, -138, -138, -138, -138,
     -138, -138, -138, -138, -138, -138, -138, -138, -138, -138,
     -138, -138, -138, -138, -138, -138,   43, -138,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -138, -138,

     -138, -138, -138, -138, -138,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -138, -138, -138, -138,   43, -138,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -138, -138, -138, -138, -138
    },

    {
        3, -139, -139, -139, -139, -139, -139, -139, -139, -139,
     -139, -139, -139, -139, -139, -139, -139, -139, -139, -139,
     -139, -139, -139, -139, -139, -139, -139, -139, -139, -139,

     -139, -139, -139, -139, -139, -139, -139, -139, -139, -139,
     -139, -139, -139, -139, -139, -139,   43, -139,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -139, -139,
     -139, -139, -139, -139, -139,   43,   43,   43,   43,   43,
       43,  183,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,  184,  185,   43,   43,   43,   43,   43,   43,
       43, -139, -139, -139, -139,   43, -139,   43,   43,   43,
       43,   43,   43,  183,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  184,  185,   43,   43,   43,   43,
       43,   43,   43, -139, -139, -139, -139, -139

    },

    {
        3, -140, -140, -140, -140, -140, -140, -140, -140, -140,
     -140, -140, -140, -140, -140, -140, -140, -140, -140, -140,
     -140, -140, -140, -140, -140, -140, -140, -140, -140, -140,
     -140, -140, -140, -140, -140, -140, -140, -140, -140, -140,
     -140, -140, -140, -140, -140, -140,   43, -140,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -140, -140,
     -140, -140, -140, -140, -140,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,  186,   43,  187,   43,   43,   43,   43,
       43, -140, -140, -140, -140,   43, -140,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,  186,   43,  187,   43,   43,
       43,   43,   43, -140, -140, -140, -140, -140
    },

```

```
    {
        3, -141, -141, -141, -141, -141, -141, -141, -141, -141,
     -141, -141, -141, -141, -141, -141, -141, -141, -141, -141,
     -141, -141, -141, -141, -141, -141, -141, -141, -141, -141,
     -141, -141, -141, -141, -141, -141, -141, -141, -141, -141,
     -141, -141, -141, -141, -141, -141,   43, -141,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -141, -141,
     -141, -141, -141, -141, -141,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -141, -141, -141, -141,   43, -141,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -141, -141, -141, -141, -141
    },

    {
        3, -142, -142, -142, -142, -142, -142, -142, -142, -142,
     -142, -142, -142, -142, -142, -142, -142, -142, -142, -142,
     -142, -142, -142, -142, -142, -142, -142, -142, -142, -142,
     -142, -142, -142, -142, -142, -142, -142, -142, -142, -142,

     -142, -142, -142, -142, -142, -142,   43, -142,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -142, -142,
     -142, -142, -142, -142, -142,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,  188,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -142, -142, -142, -142,   43, -142,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      188,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -142, -142, -142, -142, -142
    },

    {
        3, -143, -143, -143, -143, -143, -143, -143, -143, -143,

     -143, -143, -143, -143, -143, -143, -143, -143, -143, -143,
     -143, -143, -143, -143, -143, -143, -143, -143, -143, -143,
     -143, -143, -143, -143, -143, -143, -143, -143, -143, -143,
     -143, -143, -143, -143, -143, -143,   43, -143,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -143, -143,
     -143, -143, -143, -143, -143,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,  189,   43,   43,   43,   43,
       43, -143, -143, -143, -143,   43, -143,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,  189,   43,   43,
       43,   43,   43, -143, -143, -143, -143, -143
    },

    {
        3, -144, -144, -144, -144, -144, -144, -144, -144, -144,
     -144, -144, -144, -144, -144, -144, -144, -144, -144, -144,
     -144, -144, -144, -144, -144, -144, -144, -144, -144, -144,
     -144, -144, -144, -144, -144, -144, -144, -144, -144, -144,
     -144, -144, -144, -144, -144, -144,   43, -144,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -144, -144,
     -144, -144, -144, -144, -144,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,  190,   43,   43,   43,   43,   43,   43,
       43, -144, -144, -144, -144,   43, -144,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,  190,   43,   43,   43,   43,
       43,   43,   43, -144, -144, -144, -144, -144
    },

    {
        3, -145, -145, -145, -145, -145, -145, -145, -145, -145,
     -145, -145, -145, -145, -145, -145, -145, -145, -145, -145,
     -145, -145, -145, -145, -145, -145, -145, -145, -145, -145,
     -145, -145, -145, -145, -145, -145, -145, -145, -145, -145,
     -145, -145, -145, -145, -145, -145,   43, -145,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43, -145, -145,
     -145, -145, -145, -145, -145,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -145, -145, -145, -145,   43, -145,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -145, -145, -145, -145, -145
    },

    {
```

```
        3, -146, -146, -146, -146, -146, -146, -146, -146, -146,
     -146, -146, -146, -146, -146, -146, -146, -146, -146, -146,

     -146, -146, -146, -146, -146, -146, -146, -146, -146, -146,
     -146, -146, -146, -146, -146, -146, -146, -146, -146, -146,
     -146, -146, -146, -146, -146, -146,   43, -146,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -146, -146,
     -146, -146, -146, -146, -146,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  191,   43,   43,   43,   43,   43,
       43, -146, -146, -146, -146,   43, -146,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  191,   43,   43,   43,

       43,   43,   43, -146, -146, -146, -146, -146
    },

    {
        3, -147, -147, -147, -147, -147, -147, -147, -147, -147,
     -147, -147, -147, -147, -147, -147, -147, -147, -147, -147,
     -147, -147, -147, -147, -147, -147, -147, -147, -147, -147,
     -147, -147, -147, -147, -147, -147, -147, -147, -147, -147,
     -147, -147, -147, -147, -147, -147,   43, -147,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -147, -147,
     -147, -147, -147, -147, -147,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  192,   43,   43,   43,   43,   43,

       43, -147, -147, -147, -147,   43, -147,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  192,   43,   43,   43,
       43,   43,   43, -147, -147, -147, -147, -147
    },

    {
        3, -148, -148, -148, -148, -148, -148, -148, -148, -148,
     -148, -148, -148, -148, -148, -148, -148, -148, -148, -148,
     -148, -148, -148, -148, -148, -148, -148, -148, -148, -148,
     -148, -148, -148, -148, -148, -148, -148, -148, -148, -148,
     -148, -148, -148, -148, -148, -148,   43, -148,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -148, -148,

     -148, -148, -148, -148, -148,   43,   43,   43,   43,   43,
       43,  193,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -148, -148, -148, -148,   43, -148,   43,   43,   43,
       43,   43,   43,  193,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -148, -148, -148, -148, -148
    },

    {
        3, -149, -149, -149, -149, -149, -149, -149, -149, -149,
     -149, -149, -149, -149, -149, -149, -149, -149, -149, -149,
     -149, -149, -149, -149, -149, -149, -149, -149, -149, -149,

     -149, -149, -149, -149, -149, -149, -149, -149, -149, -149,
     -149, -149, -149, -149, -149, -149,   43, -149,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -149, -149,
     -149, -149, -149, -149, -149,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,  194,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -149, -149, -149, -149,   43, -149,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      194,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -149, -149, -149, -149, -149

    },

    {
        3, -150, -150, -150, -150, -150, -150, -150, -150, -150,
     -150, -150, -150, -150, -150, -150, -150, -150, -150, -150,
     -150, -150, -150, -150, -150, -150, -150, -150, -150, -150,
     -150, -150, -150, -150, -150, -150, -150, -150, -150, -150,
     -150, -150, -150, -150, -150, -150,   43, -150,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -150, -150,
     -150, -150, -150, -150, -150,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,  195,   43,   43,
       43, -150, -150, -150, -150,   43, -150,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,  195,
       43,   43,   43, -150, -150, -150, -150, -150
    },

    {
```

```
        3, -151, -151, -151, -151, -151, -151, -151, -151, -151,
     -151, -151, -151, -151, -151, -151, -151, -151, -151, -151,
     -151, -151, -151, -151, -151, -151, -151, -151, -151, -151,
     -151, -151, -151, -151, -151, -151, -151, -151, -151, -151,
     -151, -151, -151, -151, -151, -151,   43, -151,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -151, -151,
     -151, -151, -151, -151, -151,  196,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -151, -151, -151, -151,   43, -151,  196,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -151, -151, -151, -151, -151
    },

    {
        3, -152, -152, -152, -152, -152, -152, -152, -152, -152,
     -152, -152, -152, -152, -152, -152, -152, -152, -152, -152,
     -152, -152, -152, -152, -152, -152, -152, -152, -152, -152,
     -152, -152, -152, -152, -152, -152, -152, -152, -152, -152,

     -152, -152, -152, -152, -152, -152,   43, -152,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -152, -152,
     -152, -152, -152, -152, -152,   43,   43,   43,   43,  197,
       43,   43,   43,   43,   43,   43,   43,   43,  198,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -152, -152, -152, -152,   43, -152,   43,   43,   43,
       43,  197,   43,   43,   43,   43,   43,   43,   43,   43,
      198,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -152, -152, -152, -152, -152
    },

    {
        3, -153, -153, -153, -153, -153, -153, -153, -153, -153,

     -153, -153, -153, -153, -153, -153, -153, -153, -153, -153,
     -153, -153, -153, -153, -153, -153, -153, -153, -153, -153,
     -153, -153, -153, -153, -153, -153, -153, -153, -153, -153,
     -153, -153, -153, -153, -153, -153,   43, -153,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -153, -153,
     -153, -153, -153, -153, -153,   43,   43,   43,   43,   43,
      199,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -153, -153, -153, -153,   43, -153,   43,   43,   43,
       43,   43,  199,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -153, -153, -153, -153, -153
    },

    {
        3, -154, -154, -154, -154, -154, -154, -154, -154, -154,
     -154, -154, -154, -154, -154, -154, -154, -154, -154, -154,
     -154, -154, -154, -154, -154, -154, -154, -154, -154, -154,
     -154, -154, -154, -154, -154, -154, -154, -154, -154, -154,
     -154, -154, -154, -154, -154, -154,   43, -154,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -154, -154,
     -154, -154, -154, -154, -154,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -154, -154, -154, -154,   43, -154,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -154, -154, -154, -154, -154
    },

    {
        3, -155, -155, -155, -155, -155, -155, -155, -155, -155,
     -155, -155, -155, -155, -155, -155, -155, -155, -155, -155,
     -155, -155, -155, -155, -155, -155, -155, -155, -155, -155,
     -155, -155, -155, -155, -155, -155, -155, -155, -155, -155,
     -155, -155, -155, -155, -155, -155,   43, -155,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43, -155, -155,
     -155, -155, -155, -155, -155,   43,  200,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -155, -155, -155, -155,   43, -155,   43,  200,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -155, -155, -155, -155, -155
    },

    {
        3, -156, -156, -156, -156, -156, -156, -156, -156, -156,
```

```
     -156, -156, -156, -156, -156, -156, -156, -156, -156, -156,

     -156, -156, -156, -156, -156, -156, -156, -156, -156, -156,
     -156, -156, -156, -156, -156, -156, -156, -156, -156, -156,
     -156, -156, -156, -156, -156, -156,   43, -156,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -156, -156,
     -156, -156, -156, -156, -156,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -156, -156, -156, -156,   43, -156,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -156, -156, -156, -156, -156
    },

    {
        3, -157, -157, -157, -157, -157, -157, -157, -157, -157,
     -157, -157, -157, -157, -157, -157, -157, -157, -157, -157,
     -157, -157, -157, -157, -157, -157, -157, -157, -157, -157,
     -157, -157, -157, -157, -157, -157, -157, -157, -157, -157,
     -157, -157, -157, -157, -157,  201,   43, -157,  202,  202,
      202,  202,  202,  202,  202,  202,  202,  202, -157, -157,
     -157, -157, -157, -157, -157,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -157, -157, -157, -157,   43, -157,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -157, -157, -157, -157, -157
    },

    {
        3, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158,  158,  158,
      158,  158,  158,  158,  158,  158,  158,  158, -158, -158,

     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158, -158, -158,
     -158, -158, -158, -158, -158, -158, -158, -158
    },

    {
        3, -159, -159, -159, -159, -159, -159, -159, -159, -159,
     -159, -159, -159, -159, -159, -159, -159, -159, -159, -159,
     -159, -159, -159, -159, -159, -159, -159, -159, -159, -159,

     -159, -159, -159, -159, -159, -159, -159, -159, -159, -159,
     -159, -159, -159, -159, -159, -159,   43, -159,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -159, -159,
     -159, -159, -159, -159, -159,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -159, -159, -159, -159,   43, -159,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -159, -159, -159, -159, -159

    },

    {
        3, -160, -160, -160, -160, -160, -160, -160, -160, -160,
     -160, -160, -160, -160, -160, -160, -160, -160, -160, -160,
     -160, -160, -160, -160, -160, -160, -160, -160, -160, -160,
     -160, -160, -160, -160, -160, -160, -160, -160, -160, -160,
     -160, -160, -160, -160, -160, -160,   43, -160,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -160, -160,
     -160, -160, -160, -160, -160,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,  203,   43,   43,   43,   43,   43,   43,   43,
       43, -160, -160, -160, -160,   43, -160,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  203,   43,   43,   43,   43,   43,
       43,   43,   43, -160, -160, -160, -160, -160
    },

    {
        3, -161, -161, -161, -161, -161, -161, -161, -161, -161,
```

```
 -161, -161, -161, -161, -161, -161, -161, -161, -161, -161,
 -161, -161, -161, -161, -161, -161, -161, -161, -161, -161,
 -161, -161, -161, -161, -161, -161, -161, -161, -161, -161,
 -161, -161, -161, -161, -161, -161,   43, -161,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -161, -161,
 -161, -161, -161, -161, -161,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -161, -161, -161, -161,   43, -161,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -161, -161, -161, -161, -161
},

{
    3, -162, -162, -162, -162, -162, -162, -162, -162, -162,
 -162, -162, -162, -162, -162, -162, -162, -162, -162, -162,
 -162, -162, -162, -162, -162, -162, -162, -162, -162, -162,
 -162, -162, -162, -162, -162, -162, -162, -162, -162, -162,

 -162, -162, -162, -162, -162, -162,   43, -162,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -162, -162,
 -162, -162, -162, -162, -162,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -162, -162, -162, -162,  204, -162,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -162, -162, -162, -162, -162
},

{
    3, -163, -163, -163, -163, -163, -163, -163, -163, -163,

 -163, -163, -163, -163, -163, -163, -163, -163, -163, -163,
 -163, -163, -163, -163, -163, -163, -163, -163, -163, -163,
 -163, -163, -163, -163, -163, -163, -163, -163, -163, -163,
 -163, -163, -163, -163, -163, -163,   43, -163,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -163, -163,
 -163, -163, -163, -163, -163,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -163, -163, -163, -163,   43, -163,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -163, -163, -163, -163, -163
},

{
    3, -164, -164, -164, -164, -164, -164, -164, -164, -164,
 -164, -164, -164, -164, -164, -164, -164, -164, -164, -164,
 -164, -164, -164, -164, -164, -164, -164, -164, -164, -164,
 -164, -164, -164, -164, -164, -164, -164, -164, -164, -164,
 -164, -164, -164, -164, -164, -164,   43, -164,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -164, -164,
 -164, -164, -164, -164, -164,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -164, -164, -164, -164,   43, -164,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -164, -164, -164, -164, -164
},

{
    3, -165, -165, -165, -165, -165, -165, -165, -165, -165,
 -165, -165, -165, -165, -165, -165, -165, -165, -165, -165,
 -165, -165, -165, -165, -165, -165, -165, -165, -165, -165,
 -165, -165, -165, -165, -165, -165, -165, -165, -165, -165,
 -165, -165, -165, -165, -165, -165,   43, -165,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43, -165, -165,
 -165, -165, -165, -165, -165,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,  205,   43,   43,   43,   43,   43,
   43, -165, -165, -165, -165,   43, -165,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,  205,   43,   43,   43,
   43,   43,   43, -165, -165, -165, -165, -165
},

{
    3, -166, -166, -166, -166, -166, -166, -166, -166, -166,
 -166, -166, -166, -166, -166, -166, -166, -166, -166, -166,
```

```

     -166, -166, -166, -166, -166, -166, -166, -166, -166, -166,
     -166, -166, -166, -166, -166, -166, -166, -166, -166, -166,
     -166, -166, -166, -166, -166, -166,   43, -166,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -166, -166,
     -166, -166, -166, -166, -166,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -166, -166, -166, -166,   43, -166,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -166, -166, -166, -166, -166
    },

    {
        3, -167, -167, -167, -167, -167, -167, -167, -167, -167,
     -167, -167, -167, -167, -167, -167, -167, -167, -167, -167,
     -167, -167, -167, -167, -167, -167, -167, -167, -167, -167,
     -167, -167, -167, -167, -167, -167, -167, -167, -167, -167,
     -167, -167, -167, -167, -167, -167,   43, -167,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -167, -167,
     -167, -167, -167, -167, -167,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -167, -167, -167, -167,   43, -167,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -167, -167, -167, -167, -167
    },

    {
        3, -168, -168, -168, -168, -168, -168, -168, -168, -168,
     -168, -168, -168, -168, -168, -168, -168, -168, -168, -168,
     -168, -168, -168, -168, -168, -168, -168, -168, -168, -168,
     -168, -168, -168, -168, -168, -168, -168, -168, -168, -168,
     -168, -168, -168, -168, -168, -168,   43, -168,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -168, -168,

     -168, -168, -168, -168, -168,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -168, -168, -168, -168,   43, -168,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -168, -168, -168, -168, -168
    },

    {
        3, -169, -169, -169, -169, -169, -169, -169, -169, -169,
     -169, -169, -169, -169, -169, -169, -169, -169, -169, -169,
     -169, -169, -169, -169, -169, -169, -169, -169, -169, -169,

     -169, -169, -169, -169, -169, -169, -169, -169, -169, -169,
     -169, -169, -169, -169, -169, -169,   43, -169,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -169, -169,
     -169, -169, -169, -169, -169,   43,   43,   43,   43,  206,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -169, -169, -169, -169,   43, -169,   43,   43,   43,
       43,  206,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -169, -169, -169, -169, -169

    },

    {
        3, -170, -170, -170, -170, -170, -170, -170, -170, -170,
     -170, -170, -170, -170, -170, -170, -170, -170, -170, -170,
     -170, -170, -170, -170, -170, -170, -170, -170, -170, -170,
     -170, -170, -170, -170, -170, -170, -170, -170, -170, -170,
     -170, -170, -170, -170, -170, -170,   43, -170,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -170, -170,
     -170, -170, -170, -170, -170,   43,   43,   43,   43,  207,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -170, -170, -170, -170,   43, -170,   43,   43,   43,

       43,  207,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -170, -170, -170, -170, -170
    },

    {
        3, -171, -171, -171, -171, -171, -171, -171, -171, -171,
     -171, -171, -171, -171, -171, -171, -171, -171, -171, -171,
```

```
     -171, -171, -171, -171, -171, -171, -171, -171, -171, -171,
     -171, -171, -171, -171, -171, -171, -171, -171, -171, -171,
     -171, -171, -171, -171, -171, -171,   43, -171,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -171, -171,
     -171, -171, -171, -171, -171,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  208,   43,   43,   43,   43,   43,
       43, -171, -171, -171, -171,   43, -171,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  208,   43,   43,   43,
       43,   43,   43, -171, -171, -171, -171, -171
    },

    {
        3, -172, -172, -172, -172, -172, -172, -172, -172, -172,
     -172, -172, -172, -172, -172, -172, -172, -172, -172, -172,
     -172, -172, -172, -172, -172, -172, -172, -172, -172, -172,
     -172, -172, -172, -172, -172, -172, -172, -172, -172, -172,

     -172, -172, -172, -172, -172, -172,   43, -172,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -172, -172,
     -172, -172, -172, -172, -172,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -172, -172, -172, -172,   43, -172,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -172, -172, -172, -172, -172
    },

    {
        3, -173, -173, -173, -173, -173, -173, -173, -173, -173,

     -173, -173, -173, -173, -173, -173, -173, -173, -173, -173,
     -173, -173, -173, -173, -173, -173, -173, -173, -173, -173,
     -173, -173, -173, -173, -173, -173, -173, -173, -173, -173,
     -173, -173, -173, -173, -173, -173,   43, -173,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -173, -173,
     -173, -173, -173, -173, -173,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -173, -173, -173, -173,   43, -173,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -173, -173, -173, -173, -173
    },

    {
        3, -174, -174, -174, -174, -174, -174, -174, -174, -174,
     -174, -174, -174, -174, -174, -174, -174, -174, -174, -174,
     -174, -174, -174, -174, -174, -174, -174, -174, -174, -174,
     -174, -174, -174, -174, -174, -174, -174, -174, -174, -174,
     -174, -174, -174, -174, -174, -174,   43, -174,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -174, -174,
     -174, -174, -174, -174, -174,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,  209,   43,   43,   43,   43,   43,
       43, -174, -174, -174, -174,   43, -174,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  209,   43,   43,   43,
       43,   43,   43, -174, -174, -174, -174, -174
    },

    {
        3, -175, -175, -175, -175, -175, -175, -175, -175, -175,
     -175, -175, -175, -175, -175, -175, -175, -175, -175, -175,
     -175, -175, -175, -175, -175, -175, -175, -175, -175, -175,
     -175, -175, -175, -175, -175, -175, -175, -175, -175, -175,
     -175, -175, -175, -175, -175, -175,   43, -175,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43, -175, -175,
     -175, -175, -175, -175, -175,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -175, -175, -175, -175,  210, -175,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -175, -175, -175, -175, -175
    },

    {
        3, -176, -176, -176, -176, -176, -176, -176, -176, -176,
     -176, -176, -176, -176, -176, -176, -176, -176, -176, -176,

```

```
     -176, -176, -176, -176, -176, -176, -176, -176, -176, -176,
     -176, -176, -176, -176, -176, -176, -176, -176, -176, -176,
     -176, -176, -176, -176, -176, -176,   43, -176,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -176, -176,
     -176, -176, -176, -176, -176,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  211,   43,   43,   43,   43,   43,
       43, -176, -176, -176, -176,   43, -176,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  211,   43,   43,   43,

       43,   43,   43, -176, -176, -176, -176, -176
    },

    {
        3, -177, -177, -177, -177, -177, -177, -177, -177, -177,
     -177, -177, -177, -177, -177, -177, -177, -177, -177, -177,
     -177, -177, -177, -177, -177, -177, -177, -177, -177, -177,
     -177, -177, -177, -177, -177, -177, -177, -177, -177, -177,
     -177, -177, -177, -177, -177, -177,   43, -177,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -177, -177,
     -177, -177, -177, -177, -177,   43,   43,   43,   43,   43,
       43,   43,   43,  212,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -177, -177, -177, -177,   43, -177,   43,   43,   43,
       43,   43,   43,   43,   43,  212,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -177, -177, -177, -177, -177
    },

    {
        3, -178, -178, -178, -178, -178, -178, -178, -178, -178,
     -178, -178, -178, -178, -178, -178, -178, -178, -178, -178,
     -178, -178, -178, -178, -178, -178, -178, -178, -178, -178,
     -178, -178, -178, -178, -178, -178, -178, -178, -178, -178,
     -178, -178, -178, -178, -178, -178,   43, -178,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -178, -178,

     -178, -178, -178, -178, -178,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -178, -178, -178, -178,   43, -178,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -178, -178, -178, -178, -178
    },

    {
        3, -179, -179, -179, -179, -179, -179, -179, -179, -179,
     -179, -179, -179, -179, -179, -179, -179, -179, -179, -179,
     -179, -179, -179, -179, -179, -179, -179, -179, -179, -179,

     -179, -179, -179, -179, -179, -179, -179, -179, -179, -179,
     -179, -179, -179, -179, -179, -179,   43, -179,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -179, -179,
     -179, -179, -179, -179, -179,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -179, -179, -179, -179,   43, -179,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -179, -179, -179, -179, -179

    },

    {
        3, -180, -180, -180, -180, -180, -180, -180, -180, -180,
     -180, -180, -180, -180, -180, -180, -180, -180, -180, -180,
     -180, -180, -180, -180, -180, -180, -180, -180, -180, -180,
     -180, -180, -180, -180, -180, -180, -180, -180, -180, -180,
     -180, -180, -180, -180, -180, -180,   43, -180,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -180, -180,
     -180, -180, -180, -180, -180,   43,   43,   43,   43,  213,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -180, -180, -180, -180,   43, -180,   43,   43,   43,

       43,  213,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -180, -180, -180, -180, -180
    },

    {
        3, -181, -181, -181, -181, -181, -181, -181, -181, -181,
     -181, -181, -181, -181, -181, -181, -181, -181, -181, -181,
     -181, -181, -181, -181, -181, -181, -181, -181, -181, -181,
```

```
 -181, -181, -181, -181, -181, -181, -181, -181, -181, -181,
 -181, -181, -181, -181, -181, -181,   43, -181,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -181, -181,
 -181, -181, -181, -181, -181,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -181, -181, -181, -181,   43, -181,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -181, -181, -181, -181, -181
},

{
    3, -182, -182, -182, -182, -182, -182, -182, -182, -182,
 -182, -182, -182, -182, -182, -182, -182, -182, -182, -182,
 -182, -182, -182, -182, -182, -182, -182, -182, -182, -182,
 -182, -182, -182, -182, -182, -182, -182, -182, -182, -182,

 -182, -182, -182, -182, -182, -182,   43, -182,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -182, -182,
 -182, -182, -182, -182, -182,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -182, -182, -182, -182,   43, -182,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -182, -182, -182, -182, -182
},

{
    3, -183, -183, -183, -183, -183, -183, -183, -183, -183,

 -183, -183, -183, -183, -183, -183, -183, -183, -183, -183,
 -183, -183, -183, -183, -183, -183, -183, -183, -183, -183,
 -183, -183, -183, -183, -183, -183, -183, -183, -183, -183,
 -183, -183, -183, -183, -183, -183,   43, -183,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -183, -183,
 -183, -183, -183, -183, -183,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,  214,   43,   43,   43,   43,   43,   43,   43,
   43, -183, -183, -183, -183,   43, -183,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43,   43,  214,   43,   43,   43,   43,   43,
   43,   43,   43, -183, -183, -183, -183, -183
},

{
    3, -184, -184, -184, -184, -184, -184, -184, -184, -184,
 -184, -184, -184, -184, -184, -184, -184, -184, -184, -184,
 -184, -184, -184, -184, -184, -184, -184, -184, -184, -184,
 -184, -184, -184, -184, -184, -184, -184, -184, -184, -184,
 -184, -184, -184, -184, -184, -184,   43, -184,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -184, -184,
 -184, -184, -184, -184, -184,   43,   43,   43,   43,  215,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -184, -184, -184, -184,   43, -184,   43,   43,   43,
   43,  215,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -184, -184, -184, -184, -184
},

{
    3, -185, -185, -185, -185, -185, -185, -185, -185, -185,
 -185, -185, -185, -185, -185, -185, -185, -185, -185, -185,
 -185, -185, -185, -185, -185, -185, -185, -185, -185, -185,
 -185, -185, -185, -185, -185, -185, -185, -185, -185, -185,
 -185, -185, -185, -185, -185, -185,   43, -185,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43, -185, -185,
 -185, -185, -185, -185, -185,   43,   43,   43,   43,  216,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -185, -185, -185, -185,   43, -185,   43,   43,   43,
   43,  216,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -185, -185, -185, -185, -185
},

{
    3, -186, -186, -186, -186, -186, -186, -186, -186, -186,
 -186, -186, -186, -186, -186, -186, -186, -186, -186, -186,

 -186, -186, -186, -186, -186, -186, -186, -186, -186, -186,
```

```
     -186, -186, -186, -186, -186, -186, -186, -186, -186, -186,
     -186, -186, -186, -186, -186, -186,   43, -186,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -186, -186,
     -186, -186, -186, -186, -186,   43,   43,   43,   43,  217,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -186, -186, -186, -186,   43, -186,   43,   43,   43,
       43,  217,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -186, -186, -186, -186, -186
    },

    {
        3, -187, -187, -187, -187, -187, -187, -187, -187, -187,
     -187, -187, -187, -187, -187, -187, -187, -187, -187, -187,
     -187, -187, -187, -187, -187, -187, -187, -187, -187, -187,
     -187, -187, -187, -187, -187, -187, -187, -187, -187, -187,
     -187, -187, -187, -187, -187, -187,   43, -187,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -187, -187,
     -187, -187, -187, -187, -187,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  218,   43,   43,   43,   43,   43,

       43, -187, -187, -187, -187,   43, -187,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  218,   43,   43,   43,
       43,   43,   43, -187, -187, -187, -187, -187
    },

    {
        3, -188, -188, -188, -188, -188, -188, -188, -188, -188,
     -188, -188, -188, -188, -188, -188, -188, -188, -188, -188,
     -188, -188, -188, -188, -188, -188, -188, -188, -188, -188,
     -188, -188, -188, -188, -188, -188, -188, -188, -188, -188,
     -188, -188, -188, -188, -188, -188,   43, -188,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -188, -188,

     -188, -188, -188, -188, -188,   43,   43,   43,  219,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -188, -188, -188, -188,   43, -188,   43,   43,   43,
      219,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -188, -188, -188, -188, -188
    },

    {
        3, -189, -189, -189, -189, -189, -189, -189, -189, -189,
     -189, -189, -189, -189, -189, -189, -189, -189, -189, -189,
     -189, -189, -189, -189, -189, -189, -189, -189, -189, -189,

     -189, -189, -189, -189, -189, -189, -189, -189, -189, -189,
     -189, -189, -189, -189, -189, -189,   43, -189,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -189, -189,
     -189, -189, -189, -189, -189,   43,   43,   43,   43,  220,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -189, -189, -189, -189,   43, -189,   43,   43,   43,
       43,  220,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -189, -189, -189, -189, -189

    },

    {
        3, -190, -190, -190, -190, -190, -190, -190, -190, -190,
     -190, -190, -190, -190, -190, -190, -190, -190, -190, -190,
     -190, -190, -190, -190, -190, -190, -190, -190, -190, -190,
     -190, -190, -190, -190, -190, -190, -190, -190, -190, -190,
     -190, -190, -190, -190, -190, -190,   43, -190,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -190, -190,
     -190, -190, -190, -190, -190,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,  221,   43,   43,   43,   43,
       43, -190, -190, -190, -190,   43, -190,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,  221,   43,   43,
       43,   43,   43, -190, -190, -190, -190, -190
    },

    {
        3, -191, -191, -191, -191, -191, -191, -191, -191, -191,
     -191, -191, -191, -191, -191, -191, -191, -191, -191, -191,
     -191, -191, -191, -191, -191, -191, -191, -191, -191, -191,
     -191, -191, -191, -191, -191, -191, -191, -191, -191, -191,
```

```
 -191, -191, -191, -191, -191, -191,   43, -191,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -191, -191,
 -191, -191, -191, -191, -191,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -191, -191, -191, -191,   43, -191,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -191, -191, -191, -191, -191
},

{
    3, -192, -192, -192, -192, -192, -192, -192, -192, -192,
 -192, -192, -192, -192, -192, -192, -192, -192, -192, -192,
 -192, -192, -192, -192, -192, -192, -192, -192, -192, -192,
 -192, -192, -192, -192, -192, -192, -192, -192, -192, -192,

 -192, -192, -192, -192, -192, -192,   43, -192,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -192, -192,
 -192, -192, -192, -192, -192,   43,   43,   43,   43,  222,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -192, -192, -192, -192,   43, -192,   43,   43,   43,
   43,  222,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -192, -192, -192, -192, -192
},

{
    3, -193, -193, -193, -193, -193, -193, -193, -193, -193,

 -193, -193, -193, -193, -193, -193, -193, -193, -193, -193,
 -193, -193, -193, -193, -193, -193, -193, -193, -193, -193,
 -193, -193, -193, -193, -193, -193, -193, -193, -193, -193,
 -193, -193, -193, -193, -193, -193,   43, -193,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -193, -193,
 -193, -193, -193, -193, -193,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -193, -193, -193, -193,   43, -193,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -193, -193, -193, -193, -193
},

{
    3, -194, -194, -194, -194, -194, -194, -194, -194, -194,
 -194, -194, -194, -194, -194, -194, -194, -194, -194, -194,
 -194, -194, -194, -194, -194, -194, -194, -194, -194, -194,
 -194, -194, -194, -194, -194, -194, -194, -194, -194, -194,
 -194, -194, -194, -194, -194, -194,   43, -194,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43, -194, -194,
 -194, -194, -194, -194, -194,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -194, -194, -194, -194,   43, -194,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -194, -194, -194, -194, -194
},

{
    3, -195, -195, -195, -195, -195, -195, -195, -195, -195,
 -195, -195, -195, -195, -195, -195, -195, -195, -195, -195,
 -195, -195, -195, -195, -195, -195, -195, -195, -195, -195,
 -195, -195, -195, -195, -195, -195, -195, -195, -195, -195,
 -195, -195, -195, -195, -195, -195,   43, -195,   43,   43,

   43,   43,   43,   43,   43,   43,   43,   43, -195, -195,
 -195, -195, -195, -195, -195,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43, -195, -195, -195, -195,   43, -195,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   43,   43,   43, -195, -195, -195, -195, -195
},

{
    3, -196, -196, -196, -196, -196, -196, -196, -196, -196,
 -196, -196, -196, -196, -196, -196, -196, -196, -196, -196,

 -196, -196, -196, -196, -196, -196, -196, -196, -196, -196,
 -196, -196, -196, -196, -196, -196, -196, -196, -196, -196,
```

```
     -196, -196, -196, -196, -196, -196,   43, -196,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -196, -196,
     -196, -196, -196, -196, -196,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -196, -196, -196, -196,   43, -196,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43, -196, -196, -196, -196, -196
    },

    {
        3, -197, -197, -197, -197, -197, -197, -197, -197, -197,
     -197, -197, -197, -197, -197, -197, -197, -197, -197, -197,
     -197, -197, -197, -197, -197, -197, -197, -197, -197, -197,
     -197, -197, -197, -197, -197, -197, -197, -197, -197, -197,
     -197, -197, -197, -197, -197, -197,   43, -197,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -197, -197,
     -197, -197, -197, -197, -197,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -197, -197, -197, -197,   43, -197,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -197, -197, -197, -197, -197
    },

    {
        3, -198, -198, -198, -198, -198, -198, -198, -198, -198,
     -198, -198, -198, -198, -198, -198, -198, -198, -198, -198,
     -198, -198, -198, -198, -198, -198, -198, -198, -198, -198,
     -198, -198, -198, -198, -198, -198, -198, -198, -198, -198,
     -198, -198, -198, -198, -198, -198,   43, -198,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -198, -198,

     -198, -198, -198, -198, -198,   43,   43,  223,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -198, -198, -198, -198,   43, -198,   43,   43,  223,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -198, -198, -198, -198, -198
    },

    {
        3, -199, -199, -199, -199, -199, -199, -199, -199, -199,
     -199, -199, -199, -199, -199, -199, -199, -199, -199, -199,
     -199, -199, -199, -199, -199, -199, -199, -199, -199, -199,

     -199, -199, -199, -199, -199, -199, -199, -199, -199, -199,
     -199, -199, -199, -199, -199, -199,   43, -199,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -199, -199,
     -199, -199, -199, -199, -199,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -199, -199, -199, -199,   43, -199,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -199, -199, -199, -199, -199

    },

    {
        3, -200, -200, -200, -200, -200, -200, -200, -200, -200,
     -200, -200, -200, -200, -200, -200, -200, -200, -200, -200,
     -200, -200, -200, -200, -200, -200, -200, -200, -200, -200,
     -200, -200, -200, -200, -200, -200, -200, -200, -200, -200,
     -200, -200, -200, -200, -200, -200,   43, -200,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -200, -200,
     -200, -200, -200, -200, -200,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -200, -200, -200, -200,   43, -200,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -200, -200, -200, -200, -200
    },

    {
        3, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201,  224,  224,
```

```
      224,  224,  224,  224,  224,  224,  224,  224, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,

     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201, -201, -201,
     -201, -201, -201, -201, -201, -201, -201, -201
    },

    {
        3, -202, -202, -202, -202, -202, -202, -202, -202, -202,
     -202, -202, -202, -202, -202, -202, -202, -202, -202, -202,
     -202, -202, -202, -202, -202, -202, -202, -202, -202, -202,
     -202, -202, -202, -202, -202, -202, -202, -202, -202, -202,

     -202, -202, -202, -202, -202, -202,   43, -202,  202,  202,
      202,  202,  202,  202,  202,  202,  202,  202, -202, -202,
     -202, -202, -202, -202, -202,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -202, -202, -202, -202,   43, -202,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -202, -202, -202, -202, -202
    },

    {
        3, -203, -203, -203, -203, -203, -203, -203, -203, -203,

     -203, -203, -203, -203, -203, -203, -203, -203, -203, -203,
     -203, -203, -203, -203, -203, -203, -203, -203, -203, -203,
     -203, -203, -203, -203, -203, -203, -203, -203, -203, -203,
     -203, -203, -203, -203, -203, -203,   43, -203,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -203, -203,
     -203, -203, -203, -203, -203,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -203, -203, -203, -203,   43, -203,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -203, -203, -203, -203, -203
    },

    {
        3, -204, -204, -204, -204, -204, -204, -204, -204, -204,
     -204, -204, -204, -204, -204, -204, -204, -204, -204, -204,
     -204, -204, -204, -204, -204, -204, -204, -204, -204, -204,
     -204, -204, -204, -204, -204, -204, -204, -204, -204, -204,
     -204, -204, -204, -204, -204, -204,   43, -204,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -204, -204,
     -204, -204, -204, -204, -204,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43,   43,  225,   43,   43,   43,   43,   43,   43,   43,
       43, -204, -204, -204, -204,   43, -204,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  225,   43,   43,   43,   43,   43,
       43,   43,   43, -204, -204, -204, -204, -204
    },

    {
        3, -205, -205, -205, -205, -205, -205, -205, -205, -205,
     -205, -205, -205, -205, -205, -205, -205, -205, -205, -205,
     -205, -205, -205, -205, -205, -205, -205, -205, -205, -205,
     -205, -205, -205, -205, -205, -205, -205, -205, -205, -205,
     -205, -205, -205, -205, -205, -205,   43, -205,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43, -205, -205,
     -205, -205, -205, -205, -205,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,  226,   43,   43,   43,   43,   43,   43,   43,
       43, -205, -205, -205, -205,   43, -205,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  226,   43,   43,   43,   43,   43,
       43,   43,   43, -205, -205, -205, -205, -205
    },

    {
        3, -206, -206, -206, -206, -206, -206, -206, -206, -206,
     -206, -206, -206, -206, -206, -206, -206, -206, -206, -206,

     -206, -206, -206, -206, -206, -206, -206, -206, -206, -206,
     -206, -206, -206, -206, -206, -206, -206, -206, -206, -206,
     -206, -206, -206, -206, -206, -206,   43, -206,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43, -206, -206,
      -206, -206, -206, -206, -206,   43,   43,   43,  227,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -206, -206, -206, -206,   43, -206,   43,   43,   43,
       227,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43, -206, -206, -206, -206, -206
    },

    {
         3, -207, -207, -207, -207, -207, -207, -207, -207, -207,
      -207, -207, -207, -207, -207, -207, -207, -207, -207, -207,
      -207, -207, -207, -207, -207, -207, -207, -207, -207, -207,
      -207, -207, -207, -207, -207, -207, -207, -207, -207, -207,
      -207, -207, -207, -207, -207, -207,   43, -207,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -207, -207,
      -207, -207, -207, -207, -207,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43, -207, -207, -207, -207,   43, -207,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -207, -207, -207, -207, -207
    },

    {
         3, -208, -208, -208, -208, -208, -208, -208, -208, -208,
      -208, -208, -208, -208, -208, -208, -208, -208, -208, -208,
      -208, -208, -208, -208, -208, -208, -208, -208, -208, -208,
      -208, -208, -208, -208, -208, -208, -208, -208, -208, -208,
      -208, -208, -208, -208, -208, -208,   43, -208,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -208, -208,

      -208, -208, -208, -208, -208,   43,   43,   43,   43,   43,
        43,   43,   43,  228,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -208, -208, -208, -208,   43, -208,   43,   43,   43,
        43,   43,   43,   43,   43,  228,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -208, -208, -208, -208, -208
    },

    {
         3, -209, -209, -209, -209, -209, -209, -209, -209, -209,
      -209, -209, -209, -209, -209, -209, -209, -209, -209, -209,
      -209, -209, -209, -209, -209, -209, -209, -209, -209, -209,

      -209, -209, -209, -209, -209, -209, -209, -209, -209, -209,
      -209, -209, -209, -209, -209, -209,   43, -209,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -209, -209,
      -209, -209, -209, -209, -209,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -209, -209, -209, -209,   43, -209,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -209, -209, -209, -209, -209

    },

    {
         3, -210, -210, -210, -210, -210, -210, -210, -210, -210,
      -210, -210, -210, -210, -210, -210, -210, -210, -210, -210,
      -210, -210, -210, -210, -210, -210, -210, -210, -210, -210,
      -210, -210, -210, -210, -210, -210, -210, -210, -210, -210,
      -210, -210, -210, -210, -210, -210,   43, -210,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -210, -210,
      -210, -210, -210, -210, -210,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,  229,   43,   43,   43,   43,   43,   43,   43,
        43, -210, -210, -210, -210,   43, -210,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  229,   43,   43,   43,   43,   43,
        43,   43,   43, -210, -210, -210, -210, -210
    },

    {
         3, -211, -211, -211, -211, -211, -211, -211, -211, -211,
      -211, -211, -211, -211, -211, -211, -211, -211, -211, -211,
      -211, -211, -211, -211, -211, -211, -211, -211, -211, -211,
      -211, -211, -211, -211, -211, -211, -211, -211, -211, -211,
      -211, -211, -211, -211, -211, -211,   43, -211,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -211, -211,
```

```
 -211, -211, -211, -211, -211,   43,   43,   43,   43,   43,

    43,   43,   43,  230,   43,   43,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43, -211, -211, -211, -211,   43, -211,   43,   43,   43,
    43,   43,   43,   43,   43,  230,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43,   43,   43, -211, -211, -211, -211, -211
},

{
     3, -212, -212, -212, -212, -212, -212, -212, -212, -212,
 -212, -212, -212, -212, -212, -212, -212, -212, -212, -212,
 -212, -212, -212, -212, -212, -212, -212, -212, -212, -212,
 -212, -212, -212, -212, -212, -212, -212, -212, -212, -212,

 -212, -212, -212, -212, -212, -212,   43, -212,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43, -212, -212,
 -212, -212, -212, -212, -212,   43,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,  231,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43, -212, -212, -212, -212,   43, -212,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   231,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43,   43,   43, -212, -212, -212, -212, -212
},

{
     3, -213, -213, -213, -213, -213, -213, -213, -213, -213,

 -213, -213, -213, -213, -213, -213, -213, -213, -213, -213,
 -213, -213, -213, -213, -213, -213, -213, -213, -213, -213,
 -213, -213, -213, -213, -213, -213, -213, -213, -213, -213,
 -213, -213, -213, -213, -213, -213,   43, -213,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43, -213, -213,
 -213, -213, -213, -213, -213,   43,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   232,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43, -213, -213, -213, -213,   43, -213,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

    43,   43,  232,   43,   43,   43,   43,   43,   43,   43,
    43,   43,   43, -213, -213, -213, -213, -213
},

{
     3, -214, -214, -214, -214, -214, -214, -214, -214, -214,
 -214, -214, -214, -214, -214, -214, -214, -214, -214, -214,
 -214, -214, -214, -214, -214, -214, -214, -214, -214, -214,
 -214, -214, -214, -214, -214, -214, -214, -214, -214, -214,
 -214, -214, -214, -214, -214, -214,   43, -214,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43, -214, -214,
 -214, -214, -214, -214, -214,   43,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43, -214, -214, -214, -214,   43, -214,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43,   43,   43, -214, -214, -214, -214, -214
},

{
     3, -215, -215, -215, -215, -215, -215, -215, -215, -215,
 -215, -215, -215, -215, -215, -215, -215, -215, -215, -215,
 -215, -215, -215, -215, -215, -215, -215, -215, -215, -215,
 -215, -215, -215, -215, -215, -215, -215, -215, -215, -215,
 -215, -215, -215, -215, -215, -215,   43, -215,   43,   43,

    43,   43,   43,   43,   43,   43,   43,   43, -215, -215,
 -215, -215, -215, -215, -215,   43,   43,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
   233,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43, -215, -215, -215, -215,   43, -215,   43,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
    43,   43,  233,   43,   43,   43,   43,   43,   43,   43,
    43,   43,   43, -215, -215, -215, -215, -215
},

{
     3, -216, -216, -216, -216, -216, -216, -216, -216, -216,
 -216, -216, -216, -216, -216, -216, -216, -216, -216, -216,

 -216, -216, -216, -216, -216, -216, -216, -216, -216, -216,
 -216, -216, -216, -216, -216, -216, -216, -216, -216, -216,
 -216, -216, -216, -216, -216, -216,   43, -216,   43,   43,
    43,   43,   43,   43,   43,   43,   43,   43, -216, -216,
```

```
     -216, -216, -216, -216, -216,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,  234,   43,   43,   43,   43,   43,
       43, -216, -216, -216, -216,   43, -216,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,  234,   43,   43,   43,

       43,   43,   43, -216, -216, -216, -216, -216
    },

    {
        3, -217, -217, -217, -217, -217, -217, -217, -217, -217,
     -217, -217, -217, -217, -217, -217, -217, -217, -217, -217,
     -217, -217, -217, -217, -217, -217, -217, -217, -217, -217,
     -217, -217, -217, -217, -217, -217, -217, -217, -217, -217,
     -217, -217, -217, -217, -217, -217,   43, -217,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -217, -217,
     -217, -217, -217, -217, -217,   43,   43,   43,   43,   43,
       43,   43,   43,  235,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

       43, -217, -217, -217, -217,   43, -217,   43,   43,   43,
       43,   43,   43,   43,   43,  235,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -217, -217, -217, -217, -217
    },

    {
        3, -218, -218, -218, -218, -218, -218, -218, -218, -218,
     -218, -218, -218, -218, -218, -218, -218, -218, -218, -218,
     -218, -218, -218, -218, -218, -218, -218, -218, -218, -218,
     -218, -218, -218, -218, -218, -218, -218, -218, -218, -218,
     -218, -218, -218, -218, -218, -218,   43, -218,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -218, -218,

     -218, -218, -218, -218, -218,   43,   43,   43,   43,   43,
       43,   43,   43,  236,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -218, -218, -218, -218,   43, -218,   43,   43,   43,
       43,   43,   43,   43,   43,  236,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -218, -218, -218, -218, -218

    },

    {
        3, -219, -219, -219, -219, -219, -219, -219, -219, -219,
     -219, -219, -219, -219, -219, -219, -219, -219, -219, -219,
     -219, -219, -219, -219, -219, -219, -219, -219, -219, -219,

     -219, -219, -219, -219, -219, -219, -219, -219, -219, -219,
     -219, -219, -219, -219, -219, -219,   43, -219,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -219, -219,
     -219, -219, -219, -219, -219,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -219, -219, -219, -219,   43, -219,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -219, -219, -219, -219, -219

    },

    {
        3, -220, -220, -220, -220, -220, -220, -220, -220, -220,
     -220, -220, -220, -220, -220, -220, -220, -220, -220, -220,
     -220, -220, -220, -220, -220, -220, -220, -220, -220, -220,
     -220, -220, -220, -220, -220, -220, -220, -220, -220, -220,
     -220, -220, -220, -220, -220, -220,   43, -220,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -220, -220,
     -220, -220, -220, -220, -220,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43, -220, -220, -220, -220,   43, -220,   43,   43,   43,

       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       43,   43,   43, -220, -220, -220, -220, -220
    },

    {
        3, -221, -221, -221, -221, -221, -221, -221, -221, -221,
     -221, -221, -221, -221, -221, -221, -221, -221, -221, -221,
     -221, -221, -221, -221, -221, -221, -221, -221, -221, -221,
     -221, -221, -221, -221, -221, -221, -221, -221, -221, -221,
     -221, -221, -221, -221, -221, -221,   43, -221,   43,   43,
       43,   43,   43,   43,   43,   43,   43,   43, -221, -221,
     -221, -221, -221, -221, -221,   43,   43,   43,   43,   43,
```

```

      43,   43,   43,   43,   43,   43,   43,  237,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -221, -221, -221, -221,   43, -221,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,  237,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -221, -221, -221, -221, -221
    },

    {
        3, -222, -222, -222, -222, -222, -222, -222, -222, -222,
     -222, -222, -222, -222, -222, -222, -222, -222, -222, -222,
     -222, -222, -222, -222, -222, -222, -222, -222, -222, -222,
     -222, -222, -222, -222, -222, -222, -222, -222, -222, -222,

     -222, -222, -222, -222, -222, -222,   43, -222,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -222, -222,
     -222, -222, -222, -222, -222,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -222, -222, -222, -222,   43, -222,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -222, -222, -222, -222, -222
    },

    {
        3, -223, -223, -223, -223, -223, -223, -223, -223, -223,

     -223, -223, -223, -223, -223, -223, -223, -223, -223, -223,
     -223, -223, -223, -223, -223, -223, -223, -223, -223, -223,
     -223, -223, -223, -223, -223, -223, -223, -223, -223, -223,
     -223, -223, -223, -223, -223, -223,   43, -223,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -223, -223,
     -223, -223, -223, -223, -223,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -223, -223, -223, -223,   43, -223,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -223, -223, -223, -223, -223
    },

    {
        3, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224,  224,  224,
      224,  224,  224,  224,  224,  224,  224,  224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,

     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224, -224, -224,
     -224, -224, -224, -224, -224, -224, -224, -224
    },

    {
        3, -225, -225, -225, -225, -225, -225, -225, -225, -225,
     -225, -225, -225, -225, -225, -225, -225, -225, -225, -225,
     -225, -225, -225, -225, -225, -225, -225, -225, -225, -225,
     -225, -225, -225, -225, -225, -225, -225, -225, -225, -225,
     -225, -225, -225, -225, -225, -225,   43, -225,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43, -225, -225,
     -225, -225, -225, -225, -225,   43,   43,   43,   43,  238,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -225, -225, -225, -225,   43, -225,   43,   43,   43,
      43,  238,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -225, -225, -225, -225, -225
    },

    {
        3, -226, -226, -226, -226, -226, -226, -226, -226, -226,
     -226, -226, -226, -226, -226, -226, -226, -226, -226, -226,

     -226, -226, -226, -226, -226, -226, -226, -226, -226, -226,
     -226, -226, -226, -226, -226, -226, -226, -226, -226, -226,
     -226, -226, -226, -226, -226, -226,   43, -226,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -226, -226,
     -226, -226, -226, -226, -226,  239,   43,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -226, -226, -226, -226,   43, -226,  239,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43, -226, -226, -226, -226, -226
    },

    {
         3, -227, -227, -227, -227, -227, -227, -227, -227, -227,
     -227, -227, -227, -227, -227, -227, -227, -227, -227, -227,
     -227, -227, -227, -227, -227, -227, -227, -227, -227, -227,
     -227, -227, -227, -227, -227, -227, -227, -227, -227, -227,
     -227, -227, -227, -227, -227, -227,   43, -227,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -227, -227,
     -227, -227, -227, -227, -227,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43, -227, -227, -227, -227,  240, -227,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -227, -227, -227, -227, -227
    },

    {
         3, -228, -228, -228, -228, -228, -228, -228, -228, -228,
     -228, -228, -228, -228, -228, -228, -228, -228, -228, -228,
     -228, -228, -228, -228, -228, -228, -228, -228, -228, -228,
     -228, -228, -228, -228, -228, -228, -228, -228, -228, -228,
     -228, -228, -228, -228, -228, -228,   43, -228,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -228, -228,

     -228, -228, -228, -228, -228,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  241,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -228, -228, -228, -228,   43, -228,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       241,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -228, -228, -228, -228, -228
    },

    {
         3, -229, -229, -229, -229, -229, -229, -229, -229, -229,
     -229, -229, -229, -229, -229, -229, -229, -229, -229, -229,
     -229, -229, -229, -229, -229, -229, -229, -229, -229, -229,

     -229, -229, -229, -229, -229, -229, -229, -229, -229, -229,
     -229, -229, -229, -229, -229, -229,   43, -229,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -229, -229,
     -229, -229, -229, -229, -229,   43,   43,   43,   43,  242,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -229, -229, -229, -229,   43, -229,   43,   43,   43,
        43,  242,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -229, -229, -229, -229, -229

    },

    {
         3, -230, -230, -230, -230, -230, -230, -230, -230, -230,
     -230, -230, -230, -230, -230, -230, -230, -230, -230, -230,
     -230, -230, -230, -230, -230, -230, -230, -230, -230, -230,
     -230, -230, -230, -230, -230, -230, -230, -230, -230, -230,
     -230, -230, -230, -230, -230, -230,   43, -230,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -230, -230,
     -230, -230, -230, -230, -230,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,  243,   43,   43,   43,
        43, -230, -230, -230, -230,   43, -230,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  243,   43,
        43,   43,   43, -230, -230, -230, -230, -230
    },

    {
         3, -231, -231, -231, -231, -231, -231, -231, -231, -231,
     -231, -231, -231, -231, -231, -231, -231, -231, -231, -231,
     -231, -231, -231, -231, -231, -231, -231, -231, -231, -231,
     -231, -231, -231, -231, -231, -231, -231, -231, -231, -231,
     -231, -231, -231, -231, -231, -231,   43, -231,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -231, -231,
     -231, -231, -231, -231, -231,   43,   43,   43,   43,   43,

```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,  244,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -231, -231, -231, -231,   43, -231,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  244,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -231, -231, -231, -231, -231
    },

    {
         3, -232, -232, -232, -232, -232, -232, -232, -232, -232,
     -232, -232, -232, -232, -232, -232, -232, -232, -232, -232,
     -232, -232, -232, -232, -232, -232, -232, -232, -232, -232,
     -232, -232, -232, -232, -232, -232, -232, -232, -232, -232,

     -232, -232, -232, -232, -232, -232,   43, -232,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -232, -232,
     -232, -232, -232, -232, -232,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -232, -232, -232, -232,   43, -232,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -232, -232, -232, -232, -232

    },

    {
         3, -233, -233, -233, -233, -233, -233, -233, -233, -233,

     -233, -233, -233, -233, -233, -233, -233, -233, -233, -233,
     -233, -233, -233, -233, -233, -233, -233, -233, -233, -233,
     -233, -233, -233, -233, -233, -233, -233, -233, -233, -233,
     -233, -233, -233, -233, -233, -233,   43, -233,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -233, -233,
     -233, -233, -233, -233, -233,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -233, -233, -233, -233,   43, -233,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -233, -233, -233, -233, -233

    },

    {
         3, -234, -234, -234, -234, -234, -234, -234, -234, -234,
     -234, -234, -234, -234, -234, -234, -234, -234, -234, -234,
     -234, -234, -234, -234, -234, -234, -234, -234, -234, -234,
     -234, -234, -234, -234, -234, -234, -234, -234, -234, -234,
     -234, -234, -234, -234, -234, -234,   43, -234,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -234, -234,
     -234, -234, -234, -234, -234,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -234, -234, -234, -234,   43, -234,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -234, -234, -234, -234, -234

    },

    {
         3, -235, -235, -235, -235, -235, -235, -235, -235, -235,
     -235, -235, -235, -235, -235, -235, -235, -235, -235, -235,
     -235, -235, -235, -235, -235, -235, -235, -235, -235, -235,
     -235, -235, -235, -235, -235, -235, -235, -235, -235, -235,
     -235, -235, -235, -235, -235, -235,   43, -235,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43, -235, -235,
     -235, -235, -235, -235, -235,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       245, -235, -235, -235, -235,   43, -235,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,  245, -235, -235, -235, -235, -235

    },

    {
         3, -236, -236, -236, -236, -236, -236, -236, -236, -236,
     -236, -236, -236, -236, -236, -236, -236, -236, -236, -236,

     -236, -236, -236, -236, -236, -236, -236, -236, -236, -236,
     -236, -236, -236, -236, -236, -236, -236, -236, -236, -236,
     -236, -236, -236, -236, -236, -236,   43, -236,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -236, -236,
     -236, -236, -236, -236, -236,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,  246,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -236, -236, -236, -236,   43, -236,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  246,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43, -236, -236, -236, -236, -236
    },

    {
         3, -237, -237, -237, -237, -237, -237, -237, -237, -237,
     -237, -237, -237, -237, -237, -237, -237, -237, -237, -237,
     -237, -237, -237, -237, -237, -237, -237, -237, -237, -237,
     -237, -237, -237, -237, -237, -237, -237, -237, -237, -237,
     -237, -237, -237, -237, -237, -237,   43, -237,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -237, -237,
     -237, -237, -237, -237, -237,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43, -237, -237, -237, -237,   43, -237,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -237, -237, -237, -237, -237
    },

    {
         3, -238, -238, -238, -238, -238, -238, -238, -238, -238,
     -238, -238, -238, -238, -238, -238, -238, -238, -238, -238,
     -238, -238, -238, -238, -238, -238, -238, -238, -238, -238,
     -238, -238, -238, -238, -238, -238, -238, -238, -238, -238,
     -238, -238, -238, -238, -238, -238,   43, -238,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -238, -238,

     -238, -238, -238, -238, -238,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  247,   43,   43,   43,   43,   43,   43,
        43, -238, -238, -238, -238,   43, -238,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  247,   43,   43,   43,   43,
        43,   43,   43, -238, -238, -238, -238, -238
    },

    {
         3, -239, -239, -239, -239, -239, -239, -239, -239, -239,
     -239, -239, -239, -239, -239, -239, -239, -239, -239, -239,
     -239, -239, -239, -239, -239, -239, -239, -239, -239, -239,

     -239, -239, -239, -239, -239, -239, -239, -239, -239, -239,
     -239, -239, -239, -239, -239, -239,   43, -239,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -239, -239,
     -239, -239, -239, -239, -239,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,  248,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -239, -239, -239, -239,   43, -239,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
       248,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -239, -239, -239, -239, -239

    },

    {
         3, -240, -240, -240, -240, -240, -240, -240, -240, -240,
     -240, -240, -240, -240, -240, -240, -240, -240, -240, -240,
     -240, -240, -240, -240, -240, -240, -240, -240, -240, -240,
     -240, -240, -240, -240, -240, -240, -240, -240, -240, -240,
     -240, -240, -240, -240, -240, -240,   43, -240,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -240, -240,
     -240, -240, -240, -240, -240,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,  249,   43,   43,   43,   43,   43,   43,   43,
        43, -240, -240, -240, -240,   43, -240,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  249,   43,   43,   43,   43,   43,
        43,   43,   43, -240, -240, -240, -240, -240
    },

    {
         3, -241, -241, -241, -241, -241, -241, -241, -241, -241,
     -241, -241, -241, -241, -241, -241, -241, -241, -241, -241,
     -241, -241, -241, -241, -241, -241, -241, -241, -241, -241,
     -241, -241, -241, -241, -241, -241, -241, -241, -241, -241,
     -241, -241, -241, -241, -241, -241,   43, -241,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -241, -241,
     -241, -241, -241, -241, -241,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
        43,  250,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -241, -241, -241, -241,   43, -241,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  250,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -241, -241, -241, -241, -241
    },

    {
         3, -242, -242, -242, -242, -242, -242, -242, -242, -242,
     -242, -242, -242, -242, -242, -242, -242, -242, -242, -242,
     -242, -242, -242, -242, -242, -242, -242, -242, -242, -242,
     -242, -242, -242, -242, -242, -242, -242, -242, -242, -242,

     -242, -242, -242, -242, -242, -242,   43, -242,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -242, -242,
     -242, -242, -242, -242, -242,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  251,   43,   43,   43,   43,   43,   43,
        43, -242, -242, -242, -242,   43, -242,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  251,   43,   43,   43,   43,
        43,   43,   43, -242, -242, -242, -242, -242
    },

    {
         3, -243, -243, -243, -243, -243, -243, -243, -243, -243,

     -243, -243, -243, -243, -243, -243, -243, -243, -243, -243,
     -243, -243, -243, -243, -243, -243, -243, -243, -243, -243,
     -243, -243, -243, -243, -243, -243, -243, -243, -243, -243,
     -243, -243, -243, -243, -243, -243,   43, -243,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -243, -243,
     -243, -243, -243, -243, -243,   43,   43,   43,   43,  252,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -243, -243, -243, -243,   43, -243,   43,   43,   43,
        43,  252,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -243, -243, -243, -243, -243
    },

    {
         3, -244, -244, -244, -244, -244, -244, -244, -244, -244,
     -244, -244, -244, -244, -244, -244, -244, -244, -244, -244,
     -244, -244, -244, -244, -244, -244, -244, -244, -244, -244,
     -244, -244, -244, -244, -244, -244, -244, -244, -244, -244,
     -244, -244, -244, -244, -244, -244,   43, -244,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -244, -244,
     -244, -244, -244, -244, -244,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -244, -244, -244, -244,   43, -244,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -244, -244, -244, -244, -244
    },

    {
         3, -245, -245, -245, -245, -245, -245, -245, -245, -245,
     -245, -245, -245, -245, -245, -245, -245, -245, -245, -245,
     -245, -245, -245, -245, -245, -245, -245, -245, -245, -245,
     -245, -245, -245, -245, -245, -245, -245, -245, -245, -245,
     -245, -245, -245, -245, -245, -245,   43, -245,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43, -245, -245,
     -245, -245, -245, -245, -245,   43,   43,   43,   43,  253,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -245, -245, -245, -245,   43, -245,   43,   43,   43,
        43,  253,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -245, -245, -245, -245, -245
    },

    {
         3, -246, -246, -246, -246, -246, -246, -246, -246, -246,
     -246, -246, -246, -246, -246, -246, -246, -246, -246, -246,

     -246, -246, -246, -246, -246, -246, -246, -246, -246, -246,
     -246, -246, -246, -246, -246, -246, -246, -246, -246, -246,
     -246, -246, -246, -246, -246, -246,   43, -246,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -246, -246,
     -246, -246, -246, -246, -246,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
        43, -246, -246, -246, -246,   43, -246,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43,   43,   43, -246, -246, -246, -246, -246
    },

    {
         3, -247, -247, -247, -247, -247, -247, -247, -247, -247,
     -247, -247, -247, -247, -247, -247, -247, -247, -247, -247,
     -247, -247, -247, -247, -247, -247, -247, -247, -247, -247,
     -247, -247, -247, -247, -247, -247, -247, -247, -247, -247,
     -247, -247, -247, -247, -247, -247,   43, -247,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -247, -247,
     -247, -247, -247, -247, -247,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43, -247, -247, -247, -247,   43, -247,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -247, -247, -247, -247, -247
    },

    {
         3, -248, -248, -248, -248, -248, -248, -248, -248, -248,
     -248, -248, -248, -248, -248, -248, -248, -248, -248, -248,
     -248, -248, -248, -248, -248, -248, -248, -248, -248, -248,
     -248, -248, -248, -248, -248, -248, -248, -248, -248, -248,
     -248, -248, -248, -248, -248, -248,   43, -248,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -248, -248,

     -248, -248, -248, -248, -248,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  254,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -248, -248, -248, -248,   43, -248,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,  254,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -248, -248, -248, -248, -248
    },

    {
         3, -249, -249, -249, -249, -249, -249, -249, -249, -249,
     -249, -249, -249, -249, -249, -249, -249, -249, -249, -249,
     -249, -249, -249, -249, -249, -249, -249, -249, -249, -249,

     -249, -249, -249, -249, -249, -249, -249, -249, -249, -249,
     -249, -249, -249, -249, -249, -249,   43, -249,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -249, -249,
     -249, -249, -249, -249, -249,   43,   43,   43,   43,  255,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -249, -249, -249, -249,   43, -249,   43,   43,   43,
        43,  255,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -249, -249, -249, -249, -249

    },

    {
         3, -250, -250, -250, -250, -250, -250, -250, -250, -250,
     -250, -250, -250, -250, -250, -250, -250, -250, -250, -250,
     -250, -250, -250, -250, -250, -250, -250, -250, -250, -250,
     -250, -250, -250, -250, -250, -250, -250, -250, -250, -250,
     -250, -250, -250, -250, -250, -250,   43, -250,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -250, -250,
     -250, -250, -250, -250, -250,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -250, -250, -250, -250,   43, -250,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -250, -250, -250, -250, -250
    },

    {
         3, -251, -251, -251, -251, -251, -251, -251, -251, -251,
     -251, -251, -251, -251, -251, -251, -251, -251, -251, -251,
     -251, -251, -251, -251, -251, -251, -251, -251, -251, -251,
     -251, -251, -251, -251, -251, -251, -251, -251, -251, -251,
     -251, -251, -251, -251, -251, -251,   43, -251,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -251, -251,
     -251, -251, -251, -251, -251,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
```

```
      43, -251, -251, -251, -251,   43, -251,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -251, -251, -251, -251, -251
   },

   {
       3, -252, -252, -252, -252, -252, -252, -252, -252, -252,
    -252, -252, -252, -252, -252, -252, -252, -252, -252, -252,
    -252, -252, -252, -252, -252, -252, -252, -252, -252, -252,
    -252, -252, -252, -252, -252, -252, -252, -252, -252, -252,

    -252, -252, -252, -252, -252, -252,   43, -252,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -252, -252,
    -252, -252, -252, -252, -252,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -252, -252, -252, -252,  256, -252,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -252, -252, -252, -252, -252
   },

   {
       3, -253, -253, -253, -253, -253, -253, -253, -253, -253,

    -253, -253, -253, -253, -253, -253, -253, -253, -253, -253,
    -253, -253, -253, -253, -253, -253, -253, -253, -253, -253,
    -253, -253, -253, -253, -253, -253, -253, -253, -253, -253,
    -253, -253, -253, -253, -253, -253,   43, -253,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -253, -253,
    -253, -253, -253, -253, -253,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,  257,   43,   43,   43,   43,   43,   43,
      43, -253, -253, -253, -253,   43, -253,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,  257,   43,   43,   43,   43,
      43,   43,   43, -253, -253, -253, -253, -253
   },

   {
       3, -254, -254, -254, -254, -254, -254, -254, -254, -254,
    -254, -254, -254, -254, -254, -254, -254, -254, -254, -254,
    -254, -254, -254, -254, -254, -254, -254, -254, -254, -254,
    -254, -254, -254, -254, -254, -254, -254, -254, -254, -254,
    -254, -254, -254, -254, -254, -254,   43, -254,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -254, -254,
    -254, -254, -254, -254, -254,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43, -254, -254, -254, -254,   43, -254,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43, -254, -254, -254, -254, -254
   },

   {
       3, -255, -255, -255, -255, -255, -255, -255, -255, -255,
    -255, -255, -255, -255, -255, -255, -255, -255, -255, -255,
    -255, -255, -255, -255, -255, -255, -255, -255, -255, -255,
    -255, -255, -255, -255, -255, -255, -255, -255, -255, -255,
    -255, -255, -255, -255, -255, -255,   43, -255,   43,   43,

      43,   43,   43,   43,   43,   43,   43,   43, -255, -255,
    -255, -255, -255, -255, -255,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,  258,   43,   43,   43,   43,   43,   43,
      43, -255, -255, -255, -255,   43, -255,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,  258,   43,   43,   43,   43,
      43,   43,   43, -255, -255, -255, -255, -255
   },

   {
       3, -256, -256, -256, -256, -256, -256, -256, -256, -256,
    -256, -256, -256, -256, -256, -256, -256, -256, -256, -256,

    -256, -256, -256, -256, -256, -256, -256, -256, -256, -256,
    -256, -256, -256, -256, -256, -256, -256, -256, -256, -256,
    -256, -256, -256, -256, -256, -256,   43, -256,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43, -256, -256,
    -256, -256, -256, -256, -256,   43,   43,   43,   43,   43,
      43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
      43,   43,  259,   43,   43,   43,   43,   43,   43,   43,
      43, -256, -256, -256, -256,   43, -256,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,  259,   43,   43,   43,   43,   43,

        43,   43,   43, -256, -256, -256, -256, -256
    },

    {
         3, -257, -257, -257, -257, -257, -257, -257, -257, -257,
     -257, -257, -257, -257, -257, -257, -257, -257, -257, -257,
     -257, -257, -257, -257, -257, -257, -257, -257, -257, -257,
     -257, -257, -257, -257, -257, -257, -257, -257, -257, -257,
     -257, -257, -257, -257, -257, -257,   43, -257,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -257, -257,
     -257, -257, -257, -257, -257,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,

        43, -257, -257, -257, -257,   43, -257,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -257, -257, -257, -257, -257
    },

    {
         3, -258, -258, -258, -258, -258, -258, -258, -258, -258,
     -258, -258, -258, -258, -258, -258, -258, -258, -258, -258,
     -258, -258, -258, -258, -258, -258, -258, -258, -258, -258,
     -258, -258, -258, -258, -258, -258, -258, -258, -258, -258,
     -258, -258, -258, -258, -258, -258,   43, -258,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -258, -258,

     -258, -258, -258, -258, -258,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -258, -258, -258, -258,   43, -258,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -258, -258, -258, -258, -258
    },

    {
         3, -259, -259, -259, -259, -259, -259, -259, -259, -259,
     -259, -259, -259, -259, -259, -259, -259, -259, -259, -259,
     -259, -259, -259, -259, -259, -259, -259, -259, -259, -259,

     -259, -259, -259, -259, -259, -259, -259, -259, -259, -259,
     -259, -259, -259, -259, -259, -259,   43, -259,   43,   43,
        43,   43,   43,   43,   43,   43,   43, -259, -259,
     -259, -259, -259, -259, -259,   43,   43,   43,   43,  260,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -259, -259, -259, -259,   43, -259,   43,   43,   43,
        43,  260,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -259, -259, -259, -259, -259

    },

    {
         3, -260, -260, -260, -260, -260, -260, -260, -260, -260,
     -260, -260, -260, -260, -260, -260, -260, -260, -260, -260,
     -260, -260, -260, -260, -260, -260, -260, -260, -260, -260,
     -260, -260, -260, -260, -260, -260, -260, -260, -260, -260,
     -260, -260, -260, -260, -260, -260,   43, -260,   43,   43,
        43,   43,   43,   43,   43,   43,   43, -260, -260,
     -260, -260, -260, -260, -260,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,  261,   43,   43,   43,   43,   43,   43,
        43, -260, -260, -260, -260,   43, -260,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,  261,   43,   43,   43,   43,
        43,   43,   43, -260, -260, -260, -260, -260
    },

    {
         3, -261, -261, -261, -261, -261, -261, -261, -261, -261,
     -261, -261, -261, -261, -261, -261, -261, -261, -261, -261,
     -261, -261, -261, -261, -261, -261, -261, -261, -261, -261,
     -261, -261, -261, -261, -261, -261, -261, -261, -261, -261,
     -261, -261, -261, -261, -261, -261,   43, -261,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43, -261, -261,
     -261, -261, -261, -261, -261,   43,   43,   43,   43,   43,

        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43, -261, -261, -261, -261,   43, -261,   43,   43,   43,
```

```
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43,   43,   43,   43,   43,   43,   43,   43,
        43,   43,   43, -261, -261, -261, -261, -261
    },

    } ;

static yy_state_type yy_get_previous_state ( void );
static yy_state_type yy_try_NUL_trans ( yy_state_type current_state  );
static int yy_get_next_buffer ( void );
static void yynoreturn yy_fatal_error ( const char* msg  );

/* Done after the current pattern has been matched and before the
 * corresponding action - sets up yytext.
 */
#define YY_DO_BEFORE_ACTION \
    (yytext_ptr) = yy_bp; \
    yyleng = (int) (yy_cp - yy_bp); \
    (yy_hold_char) = *yy_cp; \
    *yy_cp = '\0'; \
    (yy_c_buf_p) = yy_cp;
#define YY_NUM_RULES 87
#define YY_END_OF_BUFFER 88
/* This struct is not used in this scanner,
   but its presence is necessary. */
struct yy_trans_info
    {
    flex_int32_t yy_verify;
    flex_int32_t yy_nxt;
    };
static const flex_int16_t yy_accept[262] =
    {   0,
        0,    0,   88,   86,   83,   83,   10,   11,   14,   12,
       18,   13,   85,   16,    3,    3,    8,   17,    9,   85,
       85,   85,   85,   85,   85,   85,   85,   85,   85,   85,
       85,   85,   85,   84,   85,   85,   85,   85,   19,   20,
       15,   85,   85,   85,    3,   85,   85,    4,    7,    6,
        5,   85,   85,   85,   85,   85,   85,   85,   27,   85,
       85,   85,   85,   85,   85,   85,   85,   85,   85,   85,
       23,   85,   85,   85,   85,   42,   85,   85,   85,   64,
       85,   85,   74,   63,   85,   29,   85,   85,   85,   85,
       85,   85,   85,   85,   85,   85,   85,   26,   85,   85,

       85,   85,   85,    2,    0,    3,    1,    1,   28,   85,
       85,   85,   85,   34,   85,   85,   85,   85,   85,   39,
       85,   85,   85,   25,   85,   85,   85,   85,   85,   65,
       85,   85,   41,   85,   36,   85,   85,   32,   85,   85,
       46,   85,   85,   85,   33,   85,   85,   85,   85,   85,
       85,   85,   85,   43,   85,   31,   85,    3,   78,   85,
       55,   85,   56,   24,   85,   53,   44,   47,   85,   85,
       85,   38,   52,   85,   85,   85,   85,   45,   51,   85,
       30,   48,   85,   85,   85,   85,   85,   85,   85,   85,
       40,   85,   62,   58,   57,   54,   21,   85,   49,   50,

        0,    2,   82,   85,   85,   85,   22,   85,   61,   85,
       85,   85,   85,   81,   85,   85,   85,   85,   35,   77,
       85,   66,   37,    2,   85,   85,   85,   85,   85,   85,
       85,   59,   60,   80,   85,   85,   67,   85,   85,   85,
       85,   85,   85,   75,   85,   68,   71,   85,   85,   76,
       70,   85,   85,   79,   85,   85,   69,   73,   85,   85,
       72
    } ;

static yy_state_type yy_last_accepting_state;
static char *yy_last_accepting_cpos;

static const yy_state_type yy_NUL_trans[262] =
    {   0,
        4,    4,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,

        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
```

```
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,

        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
        0
    } ;

extern int yy_flex_debug;
int yy_flex_debug = 0;

/* The intent behind this definition is that it'll catch
 * any uses of REJECT which flex missed.
 */
#define REJECT reject_used_but_not_detected
#define yymore() yymore_used_but_not_detected
#define YY_MORE_ADJ 0
#define YY_RESTORE_YY_MORE_OFFSET
char *yytext;
#line 1 "lexerparser.ll"
#line 2 "lexerparser.ll"
# include <cerrno>
# include <climits>
# include <cstdlib>
# include <string>
# include <locale>
# include <list>
# include "Genesys++-driver.h"
# include "GenesysParser.h"
# include "obj_t.h"
# include "../../util/Util.h"
# include "../../util/List.h"
# include "../ModelElement.h"
# include "../Attribute.h"
# include "../StatisticsCollector.h"
#include "../../../plugins/elements/Variable.h"
#include "../../../plugins/elements/Queue.h"
#include "../../../plugins/elements/Formula.h"
#include "../../../plugins/elements/Resource.h"
#include "../../../plugins/elements/Set.h"
// Work around an incompatibility in flex (at least versions
// 2.5.31 through 2.5.33): it generates code that does
// not conform to C89.  See Debian bug 333231
// <http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=333231>.
# undef yywrap
# define yywrap() 1

// The location of the current token.
static yy::location loc;
#line 5028 "../Genesys++-scanner.cpp"
#define YY_NO_INPUT 1
#line 48 "lexerparser.ll"
  // Code run each time a pattern is matched.
  # define YY_USER_ACTION  loc.columns (yyleng);
#line 5033 "../Genesys++-scanner.cpp"
#line 5034 "../Genesys++-scanner.cpp"

#define INITIAL 0

#ifndef YY_NO_UNISTD_H
/* Special case for "unistd.h", since it is non-ANSI. We include it way
 * down here because we want the user's section 1 to have been scanned first.
 * The user has a chance to override it with an option.
 */
#include <unistd.h>
#endif

#ifndef YY_EXTRA_TYPE
#define YY_EXTRA_TYPE void *
#endif

static int yy_init_globals ( void );

/* Accessor methods to globals.
   These are made visible to non-reentrant scanners for convenience. */

int yylex_destroy ( void );

int yyget_debug ( void );

void yyset_debug ( int debug_flag  );

YY_EXTRA_TYPE yyget_extra ( void );
```

```
void yyset_extra ( YY_EXTRA_TYPE user_defined  );

FILE *yyget_in ( void );

void yyset_in  ( FILE * _in_str  );

FILE *yyget_out ( void );

void yyset_out  ( FILE * _out_str  );

            int yyget_leng ( void );

char *yyget_text ( void );

int yyget_lineno ( void );

void yyset_lineno ( int _line_number  );

/* Macros after this point can all be overridden by user definitions in
 * section 1.
 */

#ifndef YY_SKIP_YYWRAP
#ifdef __cplusplus
extern "C" int yywrap ( void );
#else
extern int yywrap ( void );
#endif
#endif

#ifndef YY_NO_UNPUT

#endif

#ifndef yytext_ptr
static void yy_flex_strncpy ( char *, const char *, int );
#endif

#ifdef YY_NEED_STRLEN
static int yy_flex_strlen ( const char * );
#endif

#ifndef YY_NO_INPUT
#ifdef __cplusplus
static int yyinput ( void );
#else
static int input ( void );
#endif

#endif

/* Amount of stuff to slurp up with each read. */
#ifndef YY_READ_BUF_SIZE
#ifdef __ia64__
/* On IA-64, the buffer size is 16k, not 8k */
#define YY_READ_BUF_SIZE 16384
#else
#define YY_READ_BUF_SIZE 8192
#endif /* __ia64__ */
#endif

/* Copy whatever the last rule matched to the standard output. */
#ifndef ECHO
/* This used to be an fputs(), but since the string might contain NUL's,
 * we now use fwrite().
 */
#define ECHO do { if (fwrite( yytext, (size_t) yyleng, 1, yyout )) {} } while (0)
#endif

/* Gets input and stuffs it into "buf".  number of characters read, or YY_NULL,
 * is returned in "result".
 */
#ifndef YY_INPUT
#define YY_INPUT(buf,result,max_size) \
    errno=0; \
    while ( (result = (int) read( fileno(yyin), buf, (yy_size_t) max_size )) < 0 ) \
    { \
        if( errno != EINTR) \
        { \
            YY_FATAL_ERROR( "input in flex scanner failed" ); \
            break; \
        } \
        errno=0; \
        clearerr(yyin); \
    }\
\
```

```

#endif

/* No semi-colon after return; correct usage is to write "yyterminate();" -
 * we don't want an extra ';' after the "return" because that will cause
 * some compilers to complain about unreachable statements.
 */
#ifndef yyterminate
#define yyterminate() return YY_NULL
#endif

/* Number of entries by which start-condition stack grows. */
#ifndef YY_START_STACK_INCR
#define YY_START_STACK_INCR 25
#endif

/* Report a fatal error. */
#ifndef YY_FATAL_ERROR
#define YY_FATAL_ERROR(msg) yy_fatal_error( msg )
#endif

/* end tables serialization structures and prototypes */

/* Default declaration of generated scanner - a define so the user can
 * easily add parameters.
 */
#ifndef YY_DECL
#define YY_DECL_IS_OURS 1

extern int yylex (void);

#define YY_DECL int yylex (void)
#endif /* !YY_DECL */

/* Code executed at the beginning of each rule, after yytext and yyleng
 * have been set up.
 */
#ifndef YY_USER_ACTION
#define YY_USER_ACTION
#endif

/* Code executed at the end of each rule. */
#ifndef YY_BREAK
#define YY_BREAK /*LINTED*/break;
#endif

#define YY_RULE_SETUP \
    YY_USER_ACTION

YY_DECL
{
    yy_state_type yy_current_state;
    char *yy_cp, *yy_bp;
    int yy_act;

    if ( !(yy_init) )
        {
        (yy_init) = 1;

#ifdef YY_USER_INIT
        YY_USER_INIT;
#endif

        if ( ! (yy_start) )
            (yy_start) = 1; /* first start state */

        if ( ! yyin )
            yyin = stdin;

        if ( ! yyout )
            yyout = stdout;

        if ( ! YY_CURRENT_BUFFER ) {
            yyensure_buffer_stack ();
            YY_CURRENT_BUFFER_LVALUE =
                yy_create_buffer( yyin, YY_BUF_SIZE );
        }

        yy_load_buffer_state(  );
        }

    {
#line 57 "lexerparser.ll"



#line 61 "lexerparser.ll"
```

```
  // Code run each time yylex is called.
  loc.step ();



#line 5243 "../Genesys++-scanner.cpp"

    while ( /*CONSTCOND*/1 )        /* loops until end-of-file is reached */
        {
        yy_cp = (yy_c_buf_p);

        /* Support of yytext. */
        *yy_cp = (yy_hold_char);

        /* yy_bp points to the position in yy_ch_buf of the start of
         * the current run.
         */
        yy_bp = yy_cp;

        yy_current_state = (yy_start);
yy_match:
        while ( (yy_current_state = yy_nxt[yy_current_state][ YY_SC_TO_UI(*yy_cp) ]) > 0 )
            {
            if ( yy_accept[yy_current_state] )
                {
                (yy_last_accepting_state) = yy_current_state;
                (yy_last_accepting_cpos) = yy_cp;
                }

            ++yy_cp;
            }

        yy_current_state = -yy_current_state;

yy_find_action:
        yy_act = yy_accept[yy_current_state];

        YY_DO_BEFORE_ACTION;

do_action:  /* This label is used only to access EOF actions. */

        switch ( yy_act )
    { /* beginning of action switch */
            case 0: /* must back up */
            /* undo the effects of YY_DO_BEFORE_ACTION */
            *yy_cp = (yy_hold_char);
            yy_cp = (yy_last_accepting_cpos) + 1;
            yy_current_state = (yy_last_accepting_state);
            goto yy_find_action;

case 1:
YY_RULE_SETUP
#line 66 "lexerparser.ll"
{
    //Hexadecimal number
    //Will not fail because of regex
    //std::string text("Found Hexadecimal: ");
    //text += yytext;
    //driver.getModel()->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, text);
    return yy::genesyspp_parser::make_NUMH(obj_t(atof(yytext), std::string("Hexadecimal")),loc);
      }
    YY_BREAK
case 2:
YY_RULE_SETUP
#line 75 "lexerparser.ll"
{
       //Float number
       //Will not fail because of regex
       //std::string text("Found Float: ");
       //text += yytext;
       //driver.getModel()->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, text);
       return yy::genesyspp_parser::make_NUMD(obj_t(atof(yytext),std::string("Float")), loc);
     }
    YY_BREAK
case 3:
YY_RULE_SETUP
#line 84 "lexerparser.ll"
{
       //Decimal number
       //Will not fail because of regex
       //std::string text("Found Decimal: ");
       //text += yytext;
       //driver.getModel()->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, text);
       return yy::genesyspp_parser::make_NUMD(obj_t(atof(yytext),std::string("Decimal")), loc);
      }
    YY_BREAK
// special symbols
```

```
case 4:
YY_RULE_SETUP
#line 94 "lexerparser.ll"
{ return (yy::genesyspp_parser::make_oLE(obj_t(0, std::string(yytext)), loc));}
    YY_BREAK
case 5:
YY_RULE_SETUP
#line 95 "lexerparser.ll"
{ return (yy::genesyspp_parser::make_oGE(obj_t(0, std::string(yytext)), loc));}
    YY_BREAK
case 6:
YY_RULE_SETUP
#line 96 "lexerparser.ll"
{ return (yy::genesyspp_parser::make_oEQ(obj_t(0, std::string(yytext)), loc));}
    YY_BREAK
case 7:
YY_RULE_SETUP
#line 97 "lexerparser.ll"
{ return (yy::genesyspp_parser::make_oNE(obj_t(0, std::string(yytext)), loc));}
    YY_BREAK
case 8:
YY_RULE_SETUP
#line 98 "lexerparser.ll"
{return yy::genesyspp_parser::make_LESS(loc);}
    YY_BREAK
case 9:
YY_RULE_SETUP
#line 99 "lexerparser.ll"
{return yy::genesyspp_parser::make_GREATER(loc);}
    YY_BREAK
case 10:
YY_RULE_SETUP
#line 100 "lexerparser.ll"
{return yy::genesyspp_parser::make_LPAREN(loc);}
    YY_BREAK
case 11:
YY_RULE_SETUP
#line 101 "lexerparser.ll"
{return yy::genesyspp_parser::make_RPAREN(loc);}
    YY_BREAK
case 12:
YY_RULE_SETUP
#line 102 "lexerparser.ll"
{return yy::genesyspp_parser::make_PLUS(loc);}
    YY_BREAK
case 13:
YY_RULE_SETUP
#line 103 "lexerparser.ll"
{return yy::genesyspp_parser::make_MINUS(loc);}
    YY_BREAK
case 14:
YY_RULE_SETUP
#line 104 "lexerparser.ll"
{return yy::genesyspp_parser::make_STAR(loc);}
    YY_BREAK
case 15:
YY_RULE_SETUP
#line 105 "lexerparser.ll"
{return yy::genesyspp_parser::make_POWER(loc);}
    YY_BREAK
case 16:
YY_RULE_SETUP
#line 106 "lexerparser.ll"
{return yy::genesyspp_parser::make_SLASH(loc);}
    YY_BREAK
case 17:
YY_RULE_SETUP
#line 107 "lexerparser.ll"
{return yy::genesyspp_parser::make_ASSIGN(loc);}
    YY_BREAK
case 18:
YY_RULE_SETUP
#line 108 "lexerparser.ll"
{return yy::genesyspp_parser::make_COMMA(loc);}
    YY_BREAK
case 19:
YY_RULE_SETUP
#line 109 "lexerparser.ll"
{return yy::genesyspp_parser::make_LBRACKET(loc);}
    YY_BREAK
case 20:
YY_RULE_SETUP
#line 110 "lexerparser.ll"
{return yy::genesyspp_parser::make_RBRACKET(loc);}
    YY_BREAK
// boolean values
case 21:
```

```
YY_RULE_SETUP
#line 113 "lexerparser.ll"
{return yy::genesyspp_parser::make_NUMD(obj_t(1, std::string(yytext)), loc);}
    YY_BREAK
case 22:
YY_RULE_SETUP
#line 114 "lexerparser.ll"
{return yy::genesyspp_parser::make_NUMD(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// algorithmic functions
case 23:
YY_RULE_SETUP
#line 117 "lexerparser.ll"
{return yy::genesyspp_parser::make_cIF(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 24:
YY_RULE_SETUP
#line 118 "lexerparser.ll"
{return yy::genesyspp_parser::make_cELSE(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 25:
YY_RULE_SETUP
#line 119 "lexerparser.ll"
{return yy::genesyspp_parser::make_cFOR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 26:
YY_RULE_SETUP
#line 120 "lexerparser.ll"
{return yy::genesyspp_parser::make_cTO(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 27:
YY_RULE_SETUP
#line 121 "lexerparser.ll"
{return yy::genesyspp_parser::make_cDO(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// logic funcions
case 28:
YY_RULE_SETUP
#line 124 "lexerparser.ll"
{return yy::genesyspp_parser::make_oAND(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 29:
YY_RULE_SETUP
#line 125 "lexerparser.ll"
{return yy::genesyspp_parser::make_oOR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 30:
YY_RULE_SETUP
#line 126 "lexerparser.ll"
{return yy::genesyspp_parser::make_oNAND(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 31:
YY_RULE_SETUP
#line 127 "lexerparser.ll"
{return yy::genesyspp_parser::make_oXOR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 32:
YY_RULE_SETUP
#line 128 "lexerparser.ll"
{return yy::genesyspp_parser::make_oNOT(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// trigonometric functions
case 33:
YY_RULE_SETUP
#line 131 "lexerparser.ll"
{return yy::genesyspp_parser::make_fSIN(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 34:
YY_RULE_SETUP
#line 132 "lexerparser.ll"
{return yy::genesyspp_parser::make_fCOS(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// math funcions
case 35:
YY_RULE_SETUP
#line 135 "lexerparser.ll"
{return yy::genesyspp_parser::make_fROUND(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 36:
YY_RULE_SETUP
#line 136 "lexerparser.ll"
{return yy::genesyspp_parser::make_fMOD(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 37:
YY_RULE_SETUP
#line 137 "lexerparser.ll"
{return yy::genesyspp_parser::make_fTRUNC(obj_t(0, std::string(yytext)), loc);}
```

```
    YY_BREAK
case 38:
YY_RULE_SETUP
#line 138 "lexerparser.ll"
{return yy::genesyspp_parser::make_fFRAC(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 39:
YY_RULE_SETUP
#line 139 "lexerparser.ll"
{return yy::genesyspp_parser::make_fEXP(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 40:
YY_RULE_SETUP
#line 140 "lexerparser.ll"
{return yy::genesyspp_parser::make_fSQRT(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 41:
YY_RULE_SETUP
#line 141 "lexerparser.ll"
{return yy::genesyspp_parser::make_fLOG(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 42:
YY_RULE_SETUP
#line 142 "lexerparser.ll"
{return yy::genesyspp_parser::make_fLN(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// string functions
case 43:
YY_RULE_SETUP
#line 145 "lexerparser.ll"
{return yy::genesyspp_parser::make_fVAL(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 44:
YY_RULE_SETUP
#line 146 "lexerparser.ll"
{return yy::genesyspp_parser::make_fEVAL(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 45:
YY_RULE_SETUP
#line 147 "lexerparser.ll"
{return yy::genesyspp_parser::make_fLENG(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// probability distributions
case 46:
YY_RULE_SETUP
#line 150 "lexerparser.ll"
{return yy::genesyspp_parser::make_fRND1(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 47:
YY_RULE_SETUP
#line 151 "lexerparser.ll"
{return yy::genesyspp_parser::make_fEXPO(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 48:
YY_RULE_SETUP
#line 152 "lexerparser.ll"
{return yy::genesyspp_parser::make_fNORM(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 49:
YY_RULE_SETUP
#line 153 "lexerparser.ll"
{return yy::genesyspp_parser::make_fUNIF(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 50:
YY_RULE_SETUP
#line 154 "lexerparser.ll"
{return yy::genesyspp_parser::make_fWEIB(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 51:
YY_RULE_SETUP
#line 155 "lexerparser.ll"
{return yy::genesyspp_parser::make_fLOGN(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 52:
YY_RULE_SETUP
#line 156 "lexerparser.ll"
{return yy::genesyspp_parser::make_fGAMM(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 53:
YY_RULE_SETUP
#line 157 "lexerparser.ll"
{return yy::genesyspp_parser::make_fERLA(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 54:
YY_RULE_SETUP
#line 158 "lexerparser.ll"
{return yy::genesyspp_parser::make_fTRIA(obj_t(0, std::string(yytext)), loc);}
```

```
    YY_BREAK
case 55:
YY_RULE_SETUP
#line 159 "lexerparser.ll"
{return yy::genesyspp_parser::make_fBETA(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 56:
YY_RULE_SETUP
#line 160 "lexerparser.ll"
{return yy::genesyspp_parser::make_fDISC(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
// simulation infos
case 57:
YY_RULE_SETUP
#line 163 "lexerparser.ll"
{return yy::genesyspp_parser::make_fTNOW(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 58:
YY_RULE_SETUP
#line 164 "lexerparser.ll"
{return yy::genesyspp_parser::make_fTFIN(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 59:
YY_RULE_SETUP
#line 165 "lexerparser.ll"
{return yy::genesyspp_parser::make_fMAXREP(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 60:
YY_RULE_SETUP
#line 166 "lexerparser.ll"
{return yy::genesyspp_parser::make_fNUMREP(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 61:
YY_RULE_SETUP
#line 167 "lexerparser.ll"
{return yy::genesyspp_parser::make_fIDENT(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 62:
YY_RULE_SETUP
#line 169 "lexerparser.ll"
{return yy::genesyspp_parser::make_fTAVG(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
/****begin_Lexical_plugins****/
case 63:
YY_RULE_SETUP
#line 175 "lexerparser.ll"
{return yy::genesyspp_parser::make_fNR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 64:
YY_RULE_SETUP
#line 176 "lexerparser.ll"
{return yy::genesyspp_parser::make_fMR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 65:
YY_RULE_SETUP
#line 177 "lexerparser.ll"
{return yy::genesyspp_parser::make_fIRF(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 66:
YY_RULE_SETUP
#line 178 "lexerparser.ll"
{return yy::genesyspp_parser::make_fSTATE(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 67:
YY_RULE_SETUP
#line 179 "lexerparser.ll"
{return yy::genesyspp_parser::make_fSETSUM(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 68:
YY_RULE_SETUP
#line 180 "lexerparser.ll"
{return yy::genesyspp_parser::make_fRESUTIL(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 69:
YY_RULE_SETUP
#line 181 "lexerparser.ll"
{return yy::genesyspp_parser::make_fRESSEIZES(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 70:
YY_RULE_SETUP
#line 182 "lexerparser.ll"
{return yy::genesyspp_parser::make_NUMD(obj_t(-1, std::string(yytext)), loc);}
    YY_BREAK
case 71:
YY_RULE_SETUP
#line 183 "lexerparser.ll"
{return yy::genesyspp_parser::make_NUMD(obj_t(-2, std::string(yytext)), loc);}
```

```
    YY_BREAK
case 72:
YY_RULE_SETUP
#line 184 "lexerparser.ll"
{return yy::genesyspp_parser::make_NUMD(obj_t(-3, std::string(yytext)), loc);}
    YY_BREAK
case 73:
YY_RULE_SETUP
#line 185 "lexerparser.ll"
{return yy::genesyspp_parser::make_NUMD(obj_t(-4, std::string(yytext)), loc);}
    YY_BREAK
case 74:
YY_RULE_SETUP
#line 189 "lexerparser.ll"
{return yy::genesyspp_parser::make_fNQ(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 75:
YY_RULE_SETUP
#line 190 "lexerparser.ll"
{return yy::genesyspp_parser::make_fLASTINQ(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 76:
YY_RULE_SETUP
#line 191 "lexerparser.ll"
{return yy::genesyspp_parser::make_fFIRSTINQ(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 77:
YY_RULE_SETUP
#line 192 "lexerparser.ll"
{return yy::genesyspp_parser::make_fSAQUE(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 78:
YY_RULE_SETUP
#line 193 "lexerparser.ll"
{return yy::genesyspp_parser::make_fAQUE(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 79:
YY_RULE_SETUP
#line 194 "lexerparser.ll"
{return yy::genesyspp_parser::make_fENTATRANK(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 80:
YY_RULE_SETUP
#line 198 "lexerparser.ll"
{return yy::genesyspp_parser::make_fNUMSET(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 81:
YY_RULE_SETUP
#line 202 "lexerparser.ll"
{return yy::genesyspp_parser::make_fNUMGR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
case 82:
YY_RULE_SETUP
#line 203 "lexerparser.ll"
{return yy::genesyspp_parser::make_fATRGR(obj_t(0, std::string(yytext)), loc);}
    YY_BREAK
/****end_Lexical_plugins****/
case 83:
/* rule 83 can match eol */
YY_RULE_SETUP
#line 209 "lexerparser.ll"
;
    YY_BREAK
case 84:
YY_RULE_SETUP
#line 210 "lexerparser.ll"

    YY_BREAK
case 85:
YY_RULE_SETUP
#line 212 "lexerparser.ll"
{
        ModelElement* element;

        //std::cout « "Verificando o que é o Literal \""+std::string(yytext)+"\"\n";
        //std::cout « driver.getModel()->getInfos()->getName() « std::endl;
        // check if it is an ATTRIBUTE (and return the attribute ID (and not the value!)
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Attribute>(),
std::string(yytext));
        if (element != nullptr) {
            //std::cout « "Found attribute" « std::string(yytext) « std::endl;
            return yy::genesyspp_parser::make_ATRIB(obj_t(0, Util::TypeOf<Attribute>(),
element->getId()),loc);
        }

/****begin_LexicalLiterals_plugins****/

```

```
        // check VARIABLE
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), std::string(yytext));
        if (element != nullptr) { // it is a variable
            Variable* var = static_cast<Variable*>(element);
            //double variableID = var->getId();// ->getValue(); // var->getId()
        //std::cout « "FOUND VARIABLE " « var->getName() «" ID " « var->getId() « std::endl;
            return yy::genesyspp_parser::make_VARI(obj_t(0, Util::TypeOf<Variable>(), var->getId()),loc);
        }
        // check FORMULA
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), std::string(yytext));
        if (element != nullptr) { // it is a FORMULA
            Formula* form = static_cast<Formula*>(element);
            //double formulaValue = form->getValue(); // return only formula ID
        //std::cout « "FOUND FORMULA " « form->getName() «" ID " « form->getId() « std::endl;
            return yy::genesyspp_parser::make_FORM(obj_t(0, Util::TypeOf<Formula>(), form->getId()),loc);
        }
        // check QUEUE
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_QUEUE(obj_t(0, Util::TypeOf<Queue>(), element->getId()),loc);
        }
        // check RESOURCE
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_RESOURCE(obj_t(0, Util::TypeOf<Resource>(), element->getId()),loc);
        }
        //check SET
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Set>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_SET(obj_t(0, Util::TypeOf<Set>(), element->getId()),loc);
        }
/****end_LexicalLiterals_plugins****/

        //check CSTAT
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<StatisticsCollector>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_CSTAT(obj_t(0, Util::TypeOf<StatisticsCollector>(), element->getId()),loc);
        }

    // If no one before has identified this literal, then it is an ILLEGAL (not found, unknown) literal
        //Case not found retturns a illegal token
        return yy::genesyspp_parser::make_ILLEGAL(obj_t(0, std::string("Illegal")), loc);
      }
    YY_BREAK
case 86:
YY_RULE_SETUP
#line 285 "lexerparser.ll"
{return yy::genesyspp_parser::make_ILLEGAL(obj_t(1, std::string("Illegal")), loc);}
    YY_BREAK
case YY_STATE_EOF(INITIAL):
#line 287 "lexerparser.ll"
{return yy::genesyspp_parser::make_END(loc);}
    YY_BREAK
case 87:
YY_RULE_SETUP
#line 290 "lexerparser.ll"
ECHO;
    YY_BREAK
#line 5839 "../Genesys++-scanner.cpp"

    case YY_END_OF_BUFFER:
        {
        /* Amount of text matched not including the EOB char. */
        int yy_amount_of_matched_text = (int) (yy_cp - (yytext_ptr)) - 1;

        /* Undo the effects of YY_DO_BEFORE_ACTION. */
        *yy_cp = (yy_hold_char);
        YY_RESTORE_YY_MORE_OFFSET

        if ( YY_CURRENT_BUFFER_LVALUE->yy_buffer_status == YY_BUFFER_NEW )
            {
            /* We're scanning a new file or input source.  It's
             * possible that this happened because the user
             * just pointed yyin at a new source and called
```

```
             * yylex().  If so, then we have to assure
             * consistency between YY_CURRENT_BUFFER and our
             * globals.  Here is the right place to do so, because
             * this is the first action (other than possibly a
             * back-up) that will match for the new input source.
             */
            (yy_n_chars) = YY_CURRENT_BUFFER_LVALUE->yy_n_chars;
            YY_CURRENT_BUFFER_LVALUE->yy_input_file = yyin;
            YY_CURRENT_BUFFER_LVALUE->yy_buffer_status = YY_BUFFER_NORMAL;
            }

        /* Note that here we test for yy_c_buf_p "<=" to the position
         * of the first EOB in the buffer, since yy_c_buf_p will
         * already have been incremented past the NUL character
         * (since all states make transitions on EOB to the
         * end-of-buffer state).  Contrast this with the test
         * in input().
         */
        if ( (yy_c_buf_p) <= &YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[(yy_n_chars)] )
            { /* This was really a NUL. */
            yy_state_type yy_next_state;

            (yy_c_buf_p) = (yytext_ptr) + yy_amount_of_matched_text;

            yy_current_state = yy_get_previous_state(  );

            /* Okay, we're now positioned to make the NUL
             * transition.  We couldn't have
             * yy_get_previous_state() go ahead and do it
             * for us because it doesn't know how to deal
             * with the possibility of jamming (and we don't
             * want to build jamming into it because then it
             * will run more slowly).
             */

            yy_next_state = yy_try_NUL_trans( yy_current_state );

            yy_bp = (yytext_ptr) + YY_MORE_ADJ;

            if ( yy_next_state )
                {
                /* Consume the NUL. */
                yy_cp = ++(yy_c_buf_p);
                yy_current_state = yy_next_state;
                goto yy_match;
                }

            else
                {
                yy_cp = (yy_c_buf_p);
                goto yy_find_action;
                }
            }

        else switch ( yy_get_next_buffer(  ) )
            {
            case EOB_ACT_END_OF_FILE:
                {
                (yy_did_buffer_switch_on_eof) = 0;

                if ( yywrap(  ) )
                    {
                    /* Note: because we've taken care in
                     * yy_get_next_buffer() to have set up
                     * yytext, we can now set up
                     * yy_c_buf_p so that if some total
                     * hoser (like flex itself) wants to
                     * call the scanner after we return the
                     * YY_NULL, it'll still work - another
                     * YY_NULL will get returned.
                     */
                    (yy_c_buf_p) = (yytext_ptr) + YY_MORE_ADJ;

                    yy_act = YY_STATE_EOF(YY_START);
                    goto do_action;
                    }

                else
                    {
                    if ( ! (yy_did_buffer_switch_on_eof) )
                        YY_NEW_FILE;
                    }
                break;
                }

            case EOB_ACT_CONTINUE_SCAN:
                (yy_c_buf_p) =
```

```
                    (yytext_ptr) + yy_amount_of_matched_text;

                yy_current_state = yy_get_previous_state(  );

                yy_cp = (yy_c_buf_p);
                yy_bp = (yytext_ptr) + YY_MORE_ADJ;
                goto yy_match;

            case EOB_ACT_LAST_MATCH:
                (yy_c_buf_p) =
                &YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[(yy_n_chars)];

                yy_current_state = yy_get_previous_state(  );

                yy_cp = (yy_c_buf_p);
                yy_bp = (yytext_ptr) + YY_MORE_ADJ;
                goto yy_find_action;
            }
        break;
        }

    default:
        YY_FATAL_ERROR(
            "fatal flex scanner internal error--no action found" );
    } /* end of action switch */
        } /* end of scanning one token */
    } /* end of user's declarations */
} /* end of yylex */

/* yy_get_next_buffer - try to read in a new buffer
 *
 * Returns a code representing an action:
 *  EOB_ACT_LAST_MATCH -
 *  EOB_ACT_CONTINUE_SCAN - continue scanning from current position
 *  EOB_ACT_END_OF_FILE - end of file
 */
static int yy_get_next_buffer (void)
{
        char *dest = YY_CURRENT_BUFFER_LVALUE->yy_ch_buf;
    char *source = (yytext_ptr);
    int number_to_move, i;
    int ret_val;

    if ( (yy_c_buf_p) > &YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[(yy_n_chars) + 1] )
        YY_FATAL_ERROR(
        "fatal flex scanner internal error--end of buffer missed" );

    if ( YY_CURRENT_BUFFER_LVALUE->yy_fill_buffer == 0 )
        { /* Don't try to fill the buffer, so this is an EOF. */
        if ( (yy_c_buf_p) - (yytext_ptr) - YY_MORE_ADJ == 1 )
            {
            /* We matched a single character, the EOB, so
             * treat this as a final EOF.
             */
            return EOB_ACT_END_OF_FILE;
            }

        else
            {
            /* We matched some text prior to the EOB, first
             * process it.
             */
            return EOB_ACT_LAST_MATCH;
            }
        }

    /* Try to read more data. */

    /* First move last chars to start of buffer. */
    number_to_move = (int) ((yy_c_buf_p) - (yytext_ptr) - 1);

    for ( i = 0; i < number_to_move; ++i )
        *(dest++) = *(source++);

    if ( YY_CURRENT_BUFFER_LVALUE->yy_buffer_status == YY_BUFFER_EOF_PENDING )
        /* don't do the read, it's not guaranteed to return an EOF,
         * just force an EOF
         */
        YY_CURRENT_BUFFER_LVALUE->yy_n_chars = (yy_n_chars) = 0;

    else
        {
            int num_to_read =
            YY_CURRENT_BUFFER_LVALUE->yy_buf_size - number_to_move - 1;

        while ( num_to_read <= 0 )
            { /* Not enough room in the buffer - grow it. */
```

```
            /* just a shorter name for the current buffer */
            YY_BUFFER_STATE b = YY_CURRENT_BUFFER_LVALUE;

            int yy_c_buf_p_offset =
                (int) ((yy_c_buf_p) - b->yy_ch_buf);

            if ( b->yy_is_our_buffer )
                {
                int new_size = b->yy_buf_size * 2;

                if ( new_size <= 0 )
                    b->yy_buf_size += b->yy_buf_size / 8;
                else
                    b->yy_buf_size *= 2;

                b->yy_ch_buf = (char *)
                    /* Include room in for 2 EOB chars. */
                    yyrealloc( (void *) b->yy_ch_buf,
                             (yy_size_t) (b->yy_buf_size + 2)  );
                }
            else
                /* Can't grow it, we don't own it. */
                b->yy_ch_buf = NULL;

            if ( ! b->yy_ch_buf )
                YY_FATAL_ERROR(
                "fatal error - scanner input buffer overflow" );

            (yy_c_buf_p) = &b->yy_ch_buf[yy_c_buf_p_offset];

            num_to_read = YY_CURRENT_BUFFER_LVALUE->yy_buf_size -
                        number_to_move - 1;

            }

        if ( num_to_read > YY_READ_BUF_SIZE )
            num_to_read = YY_READ_BUF_SIZE;

        /* Read in more data. */
        YY_INPUT( (&YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[number_to_move]),
            (yy_n_chars), num_to_read );

        YY_CURRENT_BUFFER_LVALUE->yy_n_chars = (yy_n_chars);
        }

    if ( (yy_n_chars) == 0 )
        {
        if ( number_to_move == YY_MORE_ADJ )
            {
            ret_val = EOB_ACT_END_OF_FILE;
            yyrestart( yyin  );
            }

        else
            {
            ret_val = EOB_ACT_LAST_MATCH;
            YY_CURRENT_BUFFER_LVALUE->yy_buffer_status =
                YY_BUFFER_EOF_PENDING;
            }
        }

    else
        ret_val = EOB_ACT_CONTINUE_SCAN;

    if (((yy_n_chars) + number_to_move) > YY_CURRENT_BUFFER_LVALUE->yy_buf_size) {
        /* Extend the array by 50%, plus the number we really need. */
        int new_size = (yy_n_chars) + number_to_move + ((yy_n_chars) >> 1);
        YY_CURRENT_BUFFER_LVALUE->yy_ch_buf = (char *) yyrealloc(
            (void *) YY_CURRENT_BUFFER_LVALUE->yy_ch_buf, (yy_size_t) new_size  );
        if ( ! YY_CURRENT_BUFFER_LVALUE->yy_ch_buf )
            YY_FATAL_ERROR( "out of dynamic memory in yy_get_next_buffer()" );
        /* "- 2" to take care of EOB's */
        YY_CURRENT_BUFFER_LVALUE->yy_buf_size = (int) (new_size - 2);
    }

    (yy_n_chars) += number_to_move;
    YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[(yy_n_chars)] = YY_END_OF_BUFFER_CHAR;
    YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[(yy_n_chars) + 1] = YY_END_OF_BUFFER_CHAR;

    (yytext_ptr) = &YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[0];

    return ret_val;
}

/* yy_get_previous_state - get the state just before the EOB char was reached */
```

```
    static yy_state_type yy_get_previous_state (void)
{
    yy_state_type yy_current_state;
    char *yy_cp;

    yy_current_state = (yy_start);

    for ( yy_cp = (yytext_ptr) + YY_MORE_ADJ; yy_cp < (yy_c_buf_p); ++yy_cp )
        {
        if ( *yy_cp )
            {
            yy_current_state = yy_nxt[yy_current_state][YY_SC_TO_UI(*yy_cp)];
            }
        else
            yy_current_state = yy_NUL_trans[yy_current_state];
        if ( yy_accept[yy_current_state] )
            {
            (yy_last_accepting_state) = yy_current_state;
            (yy_last_accepting_cpos) = yy_cp;
            }
        }

    return yy_current_state;
}

/* yy_try_NUL_trans - try to make a transition on the NUL character
 *
 * synopsis
 *  next_state = yy_try_NUL_trans( current_state );
 */
    static yy_state_type yy_try_NUL_trans  (yy_state_type yy_current_state )
{
    int yy_is_jam;
        char *yy_cp = (yy_c_buf_p);

    yy_current_state = yy_NUL_trans[yy_current_state];
    yy_is_jam = (yy_current_state == 0);

    if ( ! yy_is_jam )
        {
        if ( yy_accept[yy_current_state] )
            {
            (yy_last_accepting_state) = yy_current_state;
            (yy_last_accepting_cpos) = yy_cp;
            }
        }

        return yy_is_jam ? 0 : yy_current_state;
}

#ifndef YY_NO_UNPUT

#endif

#ifndef YY_NO_INPUT
#ifdef __cplusplus
    static int yyinput (void)
#else
    static int input  (void)
#endif

{
    int c;

    *(yy_c_buf_p) = (yy_hold_char);

    if ( *(yy_c_buf_p) == YY_END_OF_BUFFER_CHAR )
        {
        /* yy_c_buf_p now points to the character we want to return.
         * If this occurs *before* the EOB characters, then it's a
         * valid NUL; if not, then we've hit the end of the buffer.
         */
        if ( (yy_c_buf_p) < &YY_CURRENT_BUFFER_LVALUE->yy_ch_buf[(yy_n_chars)] )
            /* This was really a NUL. */
            *(yy_c_buf_p) = '\0';

        else
            { /* need more input */
            int offset = (int) ((yy_c_buf_p) - (yytext_ptr));
            ++(yy_c_buf_p);

            switch ( yy_get_next_buffer(  ) )
                {
                case EOB_ACT_LAST_MATCH:
                    /* This happens because yy_g_n_b()
                     * sees that we've accumulated a
                     * token and flags that we need to
```

```
                     * try matching the token before
                     * proceeding.  But for input(),
                     * there's no matching to consider.
                     * So convert the EOB_ACT_LAST_MATCH
                     * to EOB_ACT_END_OF_FILE.
                     */

                    /* Reset buffer status. */
                    yyrestart( yyin );

                    /*FALLTHROUGH*/

                case EOB_ACT_END_OF_FILE:
                    {
                    if ( yywrap(  ) )
                        return 0;

                    if ( ! (yy_did_buffer_switch_on_eof) )
                        YY_NEW_FILE;
#ifdef __cplusplus
                    return yyinput();
#else
                    return input();
#endif
                    }

                case EOB_ACT_CONTINUE_SCAN:
                    (yy_c_buf_p) = (yytext_ptr) + offset;
                    break;
                }
            }
        }

    c = *(unsigned char *) (yy_c_buf_p);  /* cast for 8-bit char's */
    *(yy_c_buf_p) = '\0'; /* preserve yytext */
    (yy_hold_char) = *++(yy_c_buf_p);

    return c;
}
#endif  /* ifndef YY_NO_INPUT */

    void yyrestart  (FILE * input_file )
{

    if ( ! YY_CURRENT_BUFFER ){
        yyensure_buffer_stack ();
        YY_CURRENT_BUFFER_LVALUE =
            yy_create_buffer( yyin, YY_BUF_SIZE );
    }

    yy_init_buffer( YY_CURRENT_BUFFER, input_file );
    yy_load_buffer_state(  );
}

    void yy_switch_to_buffer  (YY_BUFFER_STATE  new_buffer )
{

    /* TODO. We should be able to replace this entire function body
     * with
     *      yypop_buffer_state();
     *      yypush_buffer_state(new_buffer);
     */
    yyensure_buffer_stack ();
    if ( YY_CURRENT_BUFFER == new_buffer )
        return;

    if ( YY_CURRENT_BUFFER )
        {
        /* Flush out information for old buffer. */
        *(yy_c_buf_p) = (yy_hold_char);
        YY_CURRENT_BUFFER_LVALUE->yy_buf_pos = (yy_c_buf_p);
        YY_CURRENT_BUFFER_LVALUE->yy_n_chars = (yy_n_chars);
        }

    YY_CURRENT_BUFFER_LVALUE = new_buffer;
    yy_load_buffer_state(  );

    /* We don't actually know whether we did this switch during
     * EOF (yywrap()) processing, but the only time this flag
     * is looked at is after yywrap() is called, so it's safe
     * to go ahead and always set it.
     */
    (yy_did_buffer_switch_on_eof) = 1;
}

static void yy_load_buffer_state  (void)
{
```

```
        (yy_n_chars) = YY_CURRENT_BUFFER_LVALUE->yy_n_chars;
    (yytext_ptr) = (yy_c_buf_p) = YY_CURRENT_BUFFER_LVALUE->yy_buf_pos;
    yyin = YY_CURRENT_BUFFER_LVALUE->yy_input_file;
    (yy_hold_char) = *(yy_c_buf_p);
}

    YY_BUFFER_STATE yy_create_buffer  (FILE * file, int  size )
{
    YY_BUFFER_STATE b;

    b = (YY_BUFFER_STATE) yyalloc( sizeof( struct yy_buffer_state )  );
    if ( ! b )
        YY_FATAL_ERROR( "out of dynamic memory in yy_create_buffer()" );

    b->yy_buf_size = size;

    /* yy_ch_buf has to be 2 characters longer than the size given because
     * we need to put in 2 end-of-buffer characters.
     */
    b->yy_ch_buf = (char *) yyalloc( (yy_size_t) (b->yy_buf_size + 2)  );
    if ( ! b->yy_ch_buf )
        YY_FATAL_ERROR( "out of dynamic memory in yy_create_buffer()" );

    b->yy_is_our_buffer = 1;

    yy_init_buffer( b, file );

    return b;
}

    void yy_delete_buffer (YY_BUFFER_STATE  b )
{

    if ( ! b )
        return;

    if ( b == YY_CURRENT_BUFFER ) /* Not sure if we should pop here. */
        YY_CURRENT_BUFFER_LVALUE = (YY_BUFFER_STATE) 0;

    if ( b->yy_is_our_buffer )
        yyfree( (void *) b->yy_ch_buf  );

    yyfree( (void *) b  );
}

/* Initializes or reinitializes a buffer.
 * This function is sometimes called more than once on the same buffer,
 * such as during a yyrestart() or at EOF.
 */
    static void yy_init_buffer  (YY_BUFFER_STATE  b, FILE * file )

{
    int oerrno = errno;

    yy_flush_buffer( b );

    b->yy_input_file = file;
    b->yy_fill_buffer = 1;

    /* If b is the current buffer, then yy_init_buffer was _probably_
     * called from yyrestart() or through yy_get_next_buffer.
     * In that case, we don't want to reset the lineno or column.
     */
    if (b != YY_CURRENT_BUFFER){
        b->yy_bs_lineno = 1;
        b->yy_bs_column = 0;
    }

        b->yy_is_interactive = file ? (isatty( fileno(file) ) > 0) : 0;

    errno = oerrno;
}

    void yy_flush_buffer (YY_BUFFER_STATE  b )
{
        if ( ! b )
        return;

    b->yy_n_chars = 0;

    /* We always need two end-of-buffer characters.  The first causes
     * a transition to the end-of-buffer state.  The second causes
     * a jam in that state.
     */
    b->yy_ch_buf[0] = YY_END_OF_BUFFER_CHAR;
    b->yy_ch_buf[1] = YY_END_OF_BUFFER_CHAR;

```

```
    b->yy_buf_pos = &b->yy_ch_buf[0];

    b->yy_at_bol = 1;
    b->yy_buffer_status = YY_BUFFER_NEW;

    if ( b == YY_CURRENT_BUFFER )
        yy_load_buffer_state(  );
}

void yypush_buffer_state (YY_BUFFER_STATE new_buffer )
{
        if (new_buffer == NULL)
        return;

    yyensure_buffer_stack();

    /* This block is copied from yy_switch_to_buffer. */
    if ( YY_CURRENT_BUFFER )
        {
        /* Flush out information for old buffer. */
        *(yy_c_buf_p) = (yy_hold_char);
        YY_CURRENT_BUFFER_LVALUE->yy_buf_pos = (yy_c_buf_p);
        YY_CURRENT_BUFFER_LVALUE->yy_n_chars = (yy_n_chars);
        }

    /* Only push if top exists. Otherwise, replace top. */
    if (YY_CURRENT_BUFFER)
        (yy_buffer_stack_top)++;
    YY_CURRENT_BUFFER_LVALUE = new_buffer;

    /* copied from yy_switch_to_buffer. */
    yy_load_buffer_state(  );
    (yy_did_buffer_switch_on_eof) = 1;
}

void yypop_buffer_state (void)
{
        if (!YY_CURRENT_BUFFER)
        return;

    yy_delete_buffer(YY_CURRENT_BUFFER );
    YY_CURRENT_BUFFER_LVALUE = NULL;
    if ((yy_buffer_stack_top) > 0)
        --(yy_buffer_stack_top);

    if (YY_CURRENT_BUFFER) {
        yy_load_buffer_state(  );
        (yy_did_buffer_switch_on_eof) = 1;
    }
}

/* Allocates the stack if it does not exist.
 *  Guarantees space for at least one push.
 */
static void yyensure_buffer_stack (void)
{
    yy_size_t num_to_alloc;

    if (!(yy_buffer_stack)) {

        /* First allocation is just for 2 elements, since we don't know if this
         * scanner will even need a stack. We use 2 instead of 1 to avoid an
         * immediate realloc on the next call.
         */
      num_to_alloc = 1; /* After all that talk, this was set to 1 anyways... */
        (yy_buffer_stack) = (struct yy_buffer_state**)yyalloc
                                (num_to_alloc * sizeof(struct yy_buffer_state*)
                                );
        if ( ! (yy_buffer_stack) )
            YY_FATAL_ERROR( "out of dynamic memory in yyensure_buffer_stack()" );

        memset((yy_buffer_stack), 0, num_to_alloc * sizeof(struct yy_buffer_state*));

        (yy_buffer_stack_max) = num_to_alloc;
        (yy_buffer_stack_top) = 0;
        return;
    }

    if ((yy_buffer_stack_top) >= ((yy_buffer_stack_max)) - 1){

        /* Increase the buffer to prepare for a possible push. */
        yy_size_t grow_size = 8 /* arbitrary grow size */;

        num_to_alloc = (yy_buffer_stack_max) + grow_size;
        (yy_buffer_stack) = (struct yy_buffer_state**)yyrealloc
                                ((yy_buffer_stack),
                                num_to_alloc * sizeof(struct yy_buffer_state*)
```

```
                                );
        if ( ! (yy_buffer_stack) )
            YY_FATAL_ERROR( "out of dynamic memory in yyensure_buffer_stack()" );

        /* zero only the new slots.*/
        memset((yy_buffer_stack) + (yy_buffer_stack_max), 0, grow_size * sizeof(struct yy_buffer_state*));
        (yy_buffer_stack_max) = num_to_alloc;
    }
}

YY_BUFFER_STATE yy_scan_buffer  (char * base, yy_size_t  size )
{
    YY_BUFFER_STATE b;

    if ( size < 2 ||
         base[size-2] != YY_END_OF_BUFFER_CHAR ||
         base[size-1] != YY_END_OF_BUFFER_CHAR )
        /* They forgot to leave room for the EOB's. */
        return NULL;

    b = (YY_BUFFER_STATE) yyalloc( sizeof( struct yy_buffer_state )  );
    if ( ! b )
        YY_FATAL_ERROR( "out of dynamic memory in yy_scan_buffer()" );

    b->yy_buf_size = (int) (size - 2);  /* "- 2" to take care of EOB's */
    b->yy_buf_pos = b->yy_ch_buf = base;
    b->yy_is_our_buffer = 0;
    b->yy_input_file = NULL;
    b->yy_n_chars = b->yy_buf_size;
    b->yy_is_interactive = 0;
    b->yy_at_bol = 1;
    b->yy_fill_buffer = 0;
    b->yy_buffer_status = YY_BUFFER_NEW;

    yy_switch_to_buffer( b  );

    return b;
}

YY_BUFFER_STATE yy_scan_string (const char * yystr )
{

    return yy_scan_bytes( yystr, (int) strlen(yystr) );
}

YY_BUFFER_STATE yy_scan_bytes  (const char * yybytes, int  _yybytes_len )
{
    YY_BUFFER_STATE b;
    char *buf;
    yy_size_t n;
    int i;

    /* Get memory for full buffer, including space for trailing EOB's. */
    n = (yy_size_t) (_yybytes_len + 2);
    buf = (char *) yyalloc( n  );
    if ( ! buf )
        YY_FATAL_ERROR( "out of dynamic memory in yy_scan_bytes()" );

    for ( i = 0; i < _yybytes_len; ++i )
        buf[i] = yybytes[i];

    buf[_yybytes_len] = buf[_yybytes_len+1] = YY_END_OF_BUFFER_CHAR;

    b = yy_scan_buffer( buf, n );
    if ( ! b )
        YY_FATAL_ERROR( "bad buffer in yy_scan_bytes()" );

    /* It's okay to grow etc. this buffer, and we should throw it
     * away when we're done.
     */
    b->yy_is_our_buffer = 1;

    return b;
}

#ifndef YY_EXIT_FAILURE
#define YY_EXIT_FAILURE 2
#endif

static void yynoreturn yy_fatal_error (const char* msg )
{
            fprintf( stderr, "%s\n", msg );
    exit( YY_EXIT_FAILURE );
}

/* Redefine yyless() so it works in section 3 code. */
```

```

#undef yyless
#define yyless(n) \
    do \
        { \
        /* Undo effects of setting up yytext. */ \
        int yyless_macro_arg = (n); \
        YY_LESS_LINENO(yyless_macro_arg);\
        yytext[yyleng] = (yy_hold_char); \
        (yy_c_buf_p) = yytext + yyless_macro_arg; \
        (yy_hold_char) = *(yy_c_buf_p); \
        *(yy_c_buf_p) = '\0'; \
        yyleng = yyless_macro_arg; \
        } \
    while ( 0 )

/* Accessor  methods (get/set functions) to struct members. */

int yyget_lineno  (void)
{

    return yylineno;
}

FILE *yyget_in  (void)
{
        return yyin;
}

FILE *yyget_out  (void)
{
        return yyout;
}

int yyget_leng  (void)
{
        return yyleng;
}

char *yyget_text  (void)
{
        return yytext;
}

void yyset_lineno (int  _line_number )
{

    yylineno = _line_number;
}

void yyset_in (FILE *  _in_str )
{
        yyin = _in_str ;
}

void yyset_out (FILE *  _out_str )
{
        yyout = _out_str ;
}

int yyget_debug  (void)
{
        return yy_flex_debug;
}

void yyset_debug (int  _bdebug )
{
        yy_flex_debug = _bdebug ;
}

static int yy_init_globals (void)
{
        /* Initialization is the same as for the non-reentrant scanner.
     * This function is called from yylex_destroy(), so don't allocate here.
     */

    (yy_buffer_stack) = NULL;
    (yy_buffer_stack_top) = 0;
    (yy_buffer_stack_max) = 0;
    (yy_c_buf_p) = NULL;
    (yy_init) = 0;
    (yy_start) = 0;

/* Defined in main.c */
#ifdef YY_STDINIT
    yyin = stdin;
    yyout = stdout;
```

```
#else
    yyin = NULL;
    yyout = NULL;
#endif

    /* For future reference: Set errno on error, since we are called by
     * yylex_init()
     */
    return 0;
}

/* yylex_destroy is for both reentrant and non-reentrant scanners. */
int yylex_destroy  (void)
{

    /* Pop the buffer stack, destroying each element. */
    while(YY_CURRENT_BUFFER){
        yy_delete_buffer( YY_CURRENT_BUFFER  );
        YY_CURRENT_BUFFER_LVALUE = NULL;
        yypop_buffer_state();
    }

    /* Destroy the stack itself. */
    yyfree((yy_buffer_stack) );
    (yy_buffer_stack) = NULL;

    /* Reset the globals. This is important in a non-reentrant scanner so the next time
     * yylex() is called, initialization will occur. */
    yy_init_globals( );

    return 0;
}

/*
 * Internal utility routines.
 */

#ifndef yytext_ptr
static void yy_flex_strncpy (char* s1, const char * s2, int n )
{

    int i;
    for ( i = 0; i < n; ++i )
        s1[i] = s2[i];
}
#endif

#ifdef YY_NEED_STRLEN
static int yy_flex_strlen (const char * s )
{
    int n;
    for ( n = 0; s[n]; ++n )
        ;

    return n;
}
#endif

void *yyalloc (yy_size_t  size )
{
            return malloc(size);
}

void *yyrealloc  (void * ptr, yy_size_t  size )
{

    /* The cast to (char *) in the following accommodates both
     * implementations that use char* generic pointers, and those
     * that use void* generic pointers.  It works with the latter
     * because both ANSI C and C++ allow castless assignment from
     * any pointer type to void*, and deal with argument conversions
     * as though doing an assignment.
     */
    return realloc(ptr, size);
}

void yyfree (void * ptr )
{
            free( (char *) ptr );   /* see yyrealloc() for (char *) cast */
}

#define YYTABLES_NAME "yytables"

#line 290 "lexerparser.ll"


void
```

```
genesyspp_driver::scan_begin_file ()
{
  //yy_flex_debug = trace_scanning;
  if (file.empty () || file == "-")
    yyin = stdin;
  else if (!(yyin = fopen (file.c_str (), "r")))
    {
      error ("cannot open " + file + ": " + strerror(errno));
      exit (EXIT_FAILURE);
    }
}

void genesyspp_driver::scan_begin_str ()
{
  //yy_flex_debug = trace_scanning;
  if(!str_to_parse.empty()){
    yy_scan_string (str_to_parse.c_str()); //maybe throw exception on else
  }else{
    std::string str("0");
    yy_scan_string (str.c_str()); //maybe throw exception on else
  }
}



void
genesyspp_driver::scan_end_file ()
{
  fclose (yyin);
}

void
genesyspp_driver::scan_end_str ()
{
  yy_delete_buffer(YY_CURRENT_BUFFER);
}

```

8.237 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/GenesysParser.cpp File Reference

```
#include "GenesysParser.h"
#include "Genesys++-driver.h"
```

Include dependency graph for GenesysParser.cpp:

Namespaces

- yy

Macros

- #define YY_(msgid) msgid
- #define YY_EXCEPTIONS 1
- #define YYRHSLOC(Rhs, K) ((Rhs)[K].location)
- #define YYLLOC_DEFAULT(Current, Rhs, N)
- #define YYCDEBUG if (yydebug_) (*yycdebug_)
- #define YY_SYMBOL_PRINT(Title, Symbol)

- #define YY_REDUCE_PRINT(Rule)
- #define YY_STACK_PRINT()
- #define yyerrok (yyerrstatus_ = 0)
- #define yyclearin (yyla.clear ())
- #define YYACCEPT goto yyacceptlab
- #define YYABORT goto yyabortlab
- #define YYERROR goto yyerrorlab
- #define YYRECOVERING() (!!yyerrstatus_)
- #define YYCASE_(N, S)

8.237.1 Macro Definition Documentation

8.237.1.1 YY_ `#define YY_( msgid ) msgid`

Definition at line 61 of file GenesysParser.cpp.

8.237.1.2 YY_EXCEPTIONS `#define YY_EXCEPTIONS 1`

Definition at line 71 of file GenesysParser.cpp.

8.237.1.3 YY_REDUCE_PRINT `#define YY_REDUCE_PRINT( Rule )`

Value:

```
do {                                    \
  if (yydebug_)                         \
    yy_reduce_print_ (Rule);            \
} while (false)
```

Definition at line 112 of file GenesysParser.cpp.

8.237.1.4 YY_STACK_PRINT `#define YY_STACK_PRINT( )`

Value:

```
do {                                    \
  if (yydebug_)                         \
    yy_stack_print_ ();                  \
} while (false)
```

Definition at line 118 of file GenesysParser.cpp.

8.237.1.5 YY_SYMBOL_PRINT

```
#define YY_SYMBOL_PRINT(
            Title,
            Symbol )
```

Value:

```
do {                                          \
  if (yydebug_)                               \
  {                                           \
    *yycdebug_ « Title « ' ';                \
    yy_print_ (*yycdebug_, Symbol);           \
    *yycdebug_ « '\n';                        \
  }                                           \
} while (false)
```

Definition at line 102 of file GenesysParser.cpp.

8.237.1.6 YYABORT

```
#define YYABORT goto yyabortlab
```

Definition at line 137 of file GenesysParser.cpp.

8.237.1.7 YYACCEPT

```
#define YYACCEPT goto yyacceptlab
```

Definition at line 136 of file GenesysParser.cpp.

8.237.1.8 YYCASE_

```
#define YYCASE_(
            N,
            S )
```

Value:

```
      case N:                                 \
        yyformat = S;                         \
      break
```

8.237.1.9 YYCDEBUG

```
#define YYCDEBUG if (yydebug_) (*yycdebug_)
```

Definition at line 100 of file GenesysParser.cpp.

8.237.1.10 yyclearin

```
#define yyclearin (yyla.clear ())
```

Definition at line 134 of file GenesysParser.cpp.

8.237.1.11 yyerrok `#define yyerrok (yyerrstatus_ = 0)`

Definition at line 133 of file GenesysParser.cpp.

8.237.1.12 YYERROR `#define YYERROR goto yyerrorlab`

Definition at line 138 of file GenesysParser.cpp.

8.237.1.13 YYLLOC_DEFAULT `#define YYLLOC_DEFAULT(`
Current,
Rhs,
N)

Value:

```
do                                                                  \
  if (N)                                                            \
    {                                                               \
      (Current).begin  = YYRHSLOC (Rhs, 1).begin;                   \
      (Current).end    = YYRHSLOC (Rhs, N).end;                     \
    }                                                               \
  else                                                              \
    {                                                               \
      (Current).begin = (Current).end = YYRHSLOC (Rhs, 0).end;      \
    }                                                               \
while (false)
```

Definition at line 81 of file GenesysParser.cpp.

8.237.1.14 YYRECOVERING `#define YYRECOVERING( ) (!!yyerrstatus_)`

Definition at line 139 of file GenesysParser.cpp.

8.237.1.15 YYRHSLOC `#define YYRHSLOC(`
Rhs,
K) ((Rhs)[K].location)

Definition at line 75 of file GenesysParser.cpp.

8.238 GenesysParser.cpp

```
// A Bison parser, made by GNU Bison 3.7.6.

// Skeleton implementation for Bison LALR(1) parsers in C++

// Copyright (C) 2002-2015, 2018-2021 Free Software Foundation, Inc.

// This program is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.

// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.

// You should have received a copy of the GNU General Public License
// along with this program.  If not, see <https://www.gnu.org/licenses/>.

// As a special exception, you may create a larger work that contains
// part or all of the Bison parser skeleton and distribute that work
// under terms of your choice, so long as that work isn't itself a
// parser generator using the skeleton or a modified version thereof
// as a parser skeleton.  Alternatively, if you modify or redistribute
// the parser skeleton itself, you may (at your option) remove this
// special exception, which will cause the skeleton and the resulting
// Bison output files to be licensed under the GNU General Public
// License without this special exception.

// This special exception was added by the Free Software Foundation in
// version 2.2 of Bison.

// DO NOT RELY ON FEATURES THAT ARE NOT DOCUMENTED in the manual,
// especially those whose name start with YY_ or yy_.  They are
// private implementation details that can be changed or removed.




#include "GenesysParser.h"


// Unqualified %code blocks.
#line 56 "bisonparser.yy"

# include "Genesys++-driver.h"


#line 51 "../GenesysParser.cpp"


#ifndef YY_
# if defined YYENABLE_NLS && YYENABLE_NLS
#  if ENABLE_NLS
#   include <libintl.h> // FIXME: INFRINGES ON USER NAME SPACE.
#   define YY_(msgid) dgettext ("bison-runtime", msgid)
#  endif
# endif
# ifndef YY_
#  define YY_(msgid) msgid
# endif
#endif


// Whether we are compiled with exception support.
#ifndef YY_EXCEPTIONS
# if defined __GNUC__ && !defined __EXCEPTIONS
#  define YY_EXCEPTIONS 0
# else
#  define YY_EXCEPTIONS 1
# endif
#endif

#define YYRHSLOC(Rhs, K) ((Rhs)[K].location)
/* YYLLOC_DEFAULT -- Set CURRENT to span from RHS[1] to RHS[N].
   If N is 0, then set CURRENT to the empty location which ends
   the previous symbol: RHS[0] (always defined).  */

# ifndef YYLLOC_DEFAULT
#  define YYLLOC_DEFAULT(Current, Rhs, N)                                   \
    do                                                                      \
      if (N)                                                                \
        {                                                                   \
          (Current).begin  = YYRHSLOC (Rhs, 1).begin;                       \
```

```
          (Current).end    = YYRHSLOC (Rhs, N).end;                      \
        }                                                                 \
      else                                                                \
        {                                                                 \
          (Current).begin = (Current).end = YYRHSLOC (Rhs, 0).end;       \
        }                                                                 \
    while (false)
# endif


// Enable debugging if requested.
#if YYDEBUG

// A pseudo ostream that takes yydebug_ into account.
# define YYCDEBUG if (yydebug_) (*yycdebug_)

# define YY_SYMBOL_PRINT(Title, Symbol)         \
  do {                                          \
    if (yydebug_)                               \
    {                                           \
      *yycdebug_ « Title « ' ';                \
      yy_print_ (*yycdebug_, Symbol);           \
      *yycdebug_ « '\n';                        \
    }                                           \
  } while (false)

# define YY_REDUCE_PRINT(Rule)          \
  do {                                  \
    if (yydebug_)                       \
      yy_reduce_print_ (Rule);          \
  } while (false)

# define YY_STACK_PRINT()               \
  do {                                  \
    if (yydebug_)                       \
      yy_stack_print_ ();                \
  } while (false)

#else // !YYDEBUG

# define YYCDEBUG if (false) std::cerr
# define YY_SYMBOL_PRINT(Title, Symbol)  YY_USE (Symbol)
# define YY_REDUCE_PRINT(Rule)           static_cast<void> (0)
# define YY_STACK_PRINT()                static_cast<void> (0)

#endif // !YYDEBUG

#define yyerrok         (yyerrstatus_ = 0)
#define yyclearin       (yyla.clear ())

#define YYACCEPT        goto yyacceptlab
#define YYABORT         goto yyabortlab
#define YYERROR         goto yyerrorlab
#define YYRECOVERING()  (!!yyerrstatus_)

namespace yy {
#line 143 "../GenesysParser.cpp"

  genesyspp_parser::genesyspp_parser (genesyspp_driver& driver_yyarg)
#if YYDEBUG
    : yydebug_ (false),
      yycdebug_ (&std::cerr),
#else
    :
#endif
      driver (driver_yyarg)
  {}

  genesyspp_parser::~genesyspp_parser ()
  {}

  genesyspp_parser::syntax_error::~syntax_error () YY_NOEXCEPT YY_NOTHROW
  {}

  /*---------------.
  | symbol kinds.  |
  `---------------*/



  // by_state.
  genesyspp_parser::by_state::by_state () YY_NOEXCEPT
    : state (empty_state)
  {}

  genesyspp_parser::by_state::by_state (const by_state& that) YY_NOEXCEPT
    : state (that.state)
```

```
{}

void
genesyspp_parser::by_state::clear () YY_NOEXCEPT
{
  state = empty_state;
}

void
genesyspp_parser::by_state::move (by_state& that)
{
  state = that.state;
  that.clear ();
}

genesyspp_parser::by_state::by_state (state_type s) YY_NOEXCEPT
  : state (s)
{}

genesyspp_parser::symbol_kind_type
genesyspp_parser::by_state::kind () const YY_NOEXCEPT
{
  if (state == empty_state)
    return symbol_kind::S_YYEMPTY;
  else
    return YY_CAST (symbol_kind_type, yystos_[+state]);
}

genesyspp_parser::stack_symbol_type::stack_symbol_type ()
{}

genesyspp_parser::stack_symbol_type::stack_symbol_type (YY_RVREF (stack_symbol_type) that)
  : super_type (YY_MOVE (that.state), YY_MOVE (that.location))
{
  switch (that.kind ())
  {
    case symbol_kind::S_NUMD: // NUMD
    case symbol_kind::S_NUMH: // NUMH
    case symbol_kind::S_CTEZERO: // CTEZERO
    case symbol_kind::S_oLE: // oLE
    case symbol_kind::S_oGE: // oGE
    case symbol_kind::S_oEQ: // oEQ
    case symbol_kind::S_oNE: // oNE
    case symbol_kind::S_oAND: // oAND
    case symbol_kind::S_oOR: // oOR
    case symbol_kind::S_oNAND: // oNAND
    case symbol_kind::S_oXOR: // oXOR
    case symbol_kind::S_oNOT: // oNOT
    case symbol_kind::S_fSIN: // fSIN
    case symbol_kind::S_fCOS: // fCOS
    case symbol_kind::S_fROUND: // fROUND
    case symbol_kind::S_fMOD: // fMOD
    case symbol_kind::S_fTRUNC: // fTRUNC
    case symbol_kind::S_fFRAC: // fFRAC
    case symbol_kind::S_fEXP: // fEXP
    case symbol_kind::S_fSQRT: // fSQRT
    case symbol_kind::S_fLOG: // fLOG
    case symbol_kind::S_fLN: // fLN
    case symbol_kind::S_fVAL: // fVAL
    case symbol_kind::S_fEVAL: // fEVAL
    case symbol_kind::S_fLENG: // fLENG
    case symbol_kind::S_fRND1: // fRND1
    case symbol_kind::S_fEXPO: // fEXPO
    case symbol_kind::S_fNORM: // fNORM
    case symbol_kind::S_fUNIF: // fUNIF
    case symbol_kind::S_fWEIB: // fWEIB
    case symbol_kind::S_fLOGN: // fLOGN
    case symbol_kind::S_fGAMM: // fGAMM
    case symbol_kind::S_fERLA: // fERLA
    case symbol_kind::S_fTRIA: // fTRIA
    case symbol_kind::S_fBETA: // fBETA
    case symbol_kind::S_fDISC: // fDISC
    case symbol_kind::S_fTNOW: // fTNOW
    case symbol_kind::S_fTFIN: // fTFIN
    case symbol_kind::S_fMAXREP: // fMAXREP
    case symbol_kind::S_fNUMREP: // fNUMREP
    case symbol_kind::S_fIDENT: // fIDENT
    case symbol_kind::S_cIF: // cIF
    case symbol_kind::S_cELSE: // cELSE
    case symbol_kind::S_cFOR: // cFOR
    case symbol_kind::S_cTO: // cTO
    case symbol_kind::S_cDO: // cDO
    case symbol_kind::S_ATRIB: // ATRIB
    case symbol_kind::S_CSTAT: // CSTAT
    case symbol_kind::S_fTAVG: // fTAVG
    case symbol_kind::S_ILLEGAL: // ILLEGAL
    case symbol_kind::S_RESOURCE: // RESOURCE
```

```
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.YY_MOVE_OR_COPY< obj_t > (YY_MOVE (that.value));
        break;

      default:
        break;
    }

#if 201103L <= YY_CPLUSPLUS
    // that is emptied.
    that.state = empty_state;
#endif
  }

  genesyspp_parser::stack_symbol_type::stack_symbol_type (state_type s, YY_MOVE_REF (symbol_type) that)
    : super_type (s, YY_MOVE (that.location))
  {
    switch (that.kind ())
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
```

```
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.move< obj_t > (YY_MOVE (that.value));
        break;

      default:
        break;
    }

    // that is emptied.
    that.kind_ = symbol_kind::S_YYEMPTY;
  }

#if YY_CPLUSPLUS < 201103L
  genesyspp_parser::stack_symbol_type&
  genesyspp_parser::stack_symbol_type::operator= (const stack_symbol_type& that)
  {
    state = that.state;
    switch (that.kind ())
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
```

```
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
```

```
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.copy< obj_t > (that.value);
        break;

      default:
        break;
    }

    location = that.location;
    return *this;
  }

  genesyspp_parser::stack_symbol_type&
  genesyspp_parser::stack_symbol_type::operator= (stack_symbol_type& that)
  {
    state = that.state;
    switch (that.kind ())
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
```

```
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.move< obj_t > (that.value);
        break;

      default:
        break;
    }

    location = that.location;
    // that is emptied.
    that.state = empty_state;
    return *this;
  }
#endif

  template <typename Base>
  void
  genesyspp_parser::yy_destroy_ (const char* yymsg, basic_symbol<Base>& yysym) const
  {
    if (yymsg)
      YY_SYMBOL_PRINT (yymsg, yysym);
  }

#if YYDEBUG
  template <typename Base>
  void
  genesyspp_parser::yy_print_ (std::ostream& yyo, const basic_symbol<Base>& yysym) const
  {
    std::ostream& yyoutput = yyo;
    YY_USE (yyoutput);
    if (yysym.empty ())
      yyo « "empty symbol";
    else
      {
        symbol_kind_type yykind = yysym.kind ();
        yyo « (yykind < YYNTOKENS ? "token" : "nterm")
            « ' ' « yysym.name () « " ("
            « yysym.location « ": ";
        YY_USE (yykind);
        yyo « ')';
      }
  }
#endif

  void
  genesyspp_parser::yypush_ (const char* m, YY_MOVE_REF (stack_symbol_type) sym)
  {
    if (m)
      YY_SYMBOL_PRINT (m, sym);
    yystack_.push (YY_MOVE (sym));
  }

  void
  genesyspp_parser::yypush_ (const char* m, state_type s, YY_MOVE_REF (symbol_type) sym)
  {
#if 201103L <= YY_CPLUSPLUS
    yypush_ (m, stack_symbol_type (s, std::move (sym)));
#else
    stack_symbol_type ss (s, sym);
    yypush_ (m, ss);
#endif
  }

  void
  genesyspp_parser::yypop_ (int n)
```

```
  {
    yystack_.pop (n);
  }

#if YYDEBUG
  std::ostream&
  genesyspp_parser::debug_stream () const
  {
    return *yycdebug_;
  }

  void
  genesyspp_parser::set_debug_stream (std::ostream& o)
  {
    yycdebug_ = &o;
  }


  genesyspp_parser::debug_level_type
  genesyspp_parser::debug_level () const
  {
    return yydebug_;
  }

  void
  genesyspp_parser::set_debug_level (debug_level_type l)
  {
    yydebug_ = l;
  }
#endif // YYDEBUG

  genesyspp_parser::state_type
  genesyspp_parser::yy_lr_goto_state_ (state_type yystate, int yysym)
  {
    int yyr = yypgoto_[yysym - YYNTOKENS] + yystate;
    if (0 <= yyr && yyr <= yylast_ && yycheck_[yyr] == yystate)
      return yytable_[yyr];
    else
      return yydefgoto_[yysym - YYNTOKENS];
  }

  bool
  genesyspp_parser::yy_pact_value_is_default_ (int yyvalue)
  {
    return yyvalue == yypact_ninf_;
  }

  bool
  genesyspp_parser::yy_table_value_is_error_ (int yyvalue)
  {
    return yyvalue == yytable_ninf_;
  }

  int
  genesyspp_parser::operator() ()
  {
    return parse ();
  }

  int
  genesyspp_parser::parse ()
  {
    int yyn;
    int yylen = 0;

    // Error handling.
    int yynerrs_ = 0;
    int yyerrstatus_ = 0;

    symbol_type yyla;

    stack_symbol_type yyerror_range[3];

    int yyresult;

#if YY_EXCEPTIONS
    try
#endif // YY_EXCEPTIONS
      {
    YYCDEBUG « "Starting parse\n";


    // User initialization code.
#line 49 "bisonparser.yy"
{
  // Initialize the initial location.
  //@$.begin.filename = @$.end.filename = &driver.getFile();
```

```
}

#line 789 "../GenesysParser.cpp"


    /* Initialize the stack.  The initial state will be set in
       yynewstate, since the latter expects the semantical and the
       location values to have been already stored, initialize these
       stacks with a primary value.  */
    yystack_.clear ();
    yypush_ (YY_NULLPTR, 0, YY_MOVE (yyla));

  /*--------------------------------------------.
  | yynewstate -- push a new symbol on the stack.  |
  `--------------------------------------------*/
  yynewstate:
    YYCDEBUG « "Entering state " « int (yystack_[0].state) « '\n';
    YY_STACK_PRINT ();

    // Accept?
    if (yystack_[0].state == yyfinal_)
      YYACCEPT;

    goto yybackup;


  /*-----------.
  | yybackup.  |
  `-----------*/
  yybackup:
    // Try to take a decision without lookahead.
    yyn = yypact_[+yystack_[0].state];
    if (yy_pact_value_is_default_ (yyn))
      goto yydefault;

    // Read a lookahead token.
    if (yyla.empty ())
      {
        YYCDEBUG « "Reading a token\n";
#if YY_EXCEPTIONS
        try
#endif // YY_EXCEPTIONS
          {
            symbol_type yylookahead (yylex (driver));
            yyla.move (yylookahead);
          }
#if YY_EXCEPTIONS
        catch (const syntax_error& yyexc)
          {
            YYCDEBUG « "Caught exception: " « yyexc.what() « '\n';
            error (yyexc);
            goto yyerrlab1;
          }
#endif // YY_EXCEPTIONS
      }
    YY_SYMBOL_PRINT ("Next token is", yyla);

    if (yyla.kind () == symbol_kind::S_YYerror)
    {
      // The scanner already issued an error message, process directly
      // to error recovery.  But do not keep the error token as
      // lookahead, it is too special and may lead us to an endless
      // loop in error recovery. */
      yyla.kind_ = symbol_kind::S_YYUNDEF;
      goto yyerrlab1;
    }

    /* If the proper action on seeing token YYLA.TYPE is to reduce or
       to detect an error, take that action.  */
    yyn += yyla.kind ();
    if (yyn < 0 || yylast_ < yyn || yycheck_[yyn] != yyla.kind ())
      {
        goto yydefault;
      }

    // Reduce or error.
    yyn = yytable_[yyn];
    if (yyn <= 0)
      {
        if (yy_table_value_is_error_ (yyn))
          goto yyerrlab;
        yyn = -yyn;
        goto yyreduce;
      }

    // Count tokens shifted since error; after three, turn off error status.
    if (yyerrstatus_)
```

```
      --yyerrstatus_;

    // Shift the lookahead token.
    yypush_ ("Shifting", state_type (yyn), YY_MOVE (yyla));
    goto yynewstate;


  /*-----------------------------------------------------------.
  | yydefault -- do the default action for the current state.  |
  `-----------------------------------------------------------*/
  yydefault:
    yyn = yydefact_[+yystack_[0].state];
    if (yyn == 0)
      goto yyerrlab;
    goto yyreduce;


  /*-----------------------------.
  | yyreduce -- do a reduction.  |
  `-----------------------------*/
  yyreduce:
    yylen = yyr2_[yyn];
    {
      stack_symbol_type yylhs;
      yylhs.state = yy_lr_goto_state_ (yystack_[yylen].state, yyr1_[yyn]);
      /* Variants are always initialized to an empty instance of the
         correct type. The default '$$ = $1' action is NOT applied
         when using variants.  */
      switch (yyr1_[yyn])
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
```

```
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        yylhs.value.emplace< obj_t > ();
        break;

      default:
        break;
    }


      // Default location.
      {
        stack_type::slice range (yystack_, yylen);
        YYLLOC_DEFAULT (yylhs.location, range, yylen);
        yyerror_range[1].location = yylhs.location;
      }

      // Perform the reduction.
      YY_REDUCE_PRINT (yyn);
#if YY_EXCEPTIONS
      try
#endif // YY_EXCEPTIONS
        {
          switch (yyn)
            {
  case 2: // input: expression
#line 225 "bisonparser.yy"
                    { driver.setResult(yystack_[0].value.as < obj_t > ().valor);}
#line 1022 "../GenesysParser.cpp"
    break;

  case 3: // input: error '\n'
#line 226 "bisonparser.yy"
                        { yyerrok; }
#line 1028 "../GenesysParser.cpp"
    break;

  case 4: // expression: numero
#line 230 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1034 "../GenesysParser.cpp"
    break;

  case 5: // expression: function
#line 231 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1040 "../GenesysParser.cpp"
    break;

  case 6: // expression: comando
#line 232 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as <
```

```
obj_t > ().valor;}
#line 1046 "../GenesysParser.cpp"
    break;

  case 7: // expression: atribuicao
#line 233 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1052 "../GenesysParser.cpp"
    break;

  case 8: // expression: aritmetica
#line 234 "bisonparser.yy"
                                        {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1058 "../GenesysParser.cpp"
    break;

  case 9: // expression: logica
#line 235 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1064 "../GenesysParser.cpp"
    break;

  case 10: // expression: relacional
#line 236 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1070 "../GenesysParser.cpp"
    break;

  case 11: // expression: "(" expression ")"
#line 237 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[1].value.as < obj_t > ().valor;}
#line 1076 "../GenesysParser.cpp"
    break;

  case 12: // expression: atributo
#line 238 "bisonparser.yy"
                                    {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1082 "../GenesysParser.cpp"
    break;

  case 13: // expression: variavel
#line 243 "bisonparser.yy"
                                            {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1088 "../GenesysParser.cpp"
    break;

  case 14: // expression: formula
#line 247 "bisonparser.yy"
                                            {yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1094 "../GenesysParser.cpp"
    break;

  case 15: // numero: NUMD
#line 254 "bisonparser.yy"
              { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1100 "../GenesysParser.cpp"
    break;

  case 16: // numero: NUMH
#line 255 "bisonparser.yy"
              { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1106 "../GenesysParser.cpp"
    break;

  case 17: // aritmetica: expression "+" expression
#line 259 "bisonparser.yy"
                                  { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor + yystack_[0].value.as < obj_t > ().valor;}
#line 1112 "../GenesysParser.cpp"
    break;

  case 18: // aritmetica: expression "-" expression
#line 260 "bisonparser.yy"
                                  { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor - yystack_[0].value.as < obj_t > ().valor;}
#line 1118 "../GenesysParser.cpp"
    break;

  case 19: // aritmetica: expression "/" expression
```

```
#line 261 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor / yystack_[0].value.as < obj_t > ().valor;}
#line 1124 "../GenesysParser.cpp"
    break;

  case 20: // aritmetica: expression "*" expression
#line 262 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor * yystack_[0].value.as < obj_t > ().valor;}
#line 1130 "../GenesysParser.cpp"
    break;

  case 21: // aritmetica: expression "^" expression
#line 263 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = pow(yystack_[2].value.as < obj_t > ().valor,yystack_[0].value.as < obj_t > ().valor);}
#line 1136 "../GenesysParser.cpp"
    break;

  case 22: // aritmetica: "-" expression
#line 264 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = -yystack_[0].value.as < obj_t > ().valor;}
#line 1142 "../GenesysParser.cpp"
    break;

  case 23: // logica: expression oAND expression
#line 268 "bisonparser.yy"
                                   { yylhs.value.as < obj_t > ().valor = (int) yystack_[2].value.as < obj_t > ().valor && (int) yystack_[0].value.as < obj_t > ().valor;}
#line 1148 "../GenesysParser.cpp"
    break;

  case 24: // logica: expression oOR expression
#line 269 "bisonparser.yy"
                                   { yylhs.value.as < obj_t > ().valor = (int) yystack_[2].value.as < obj_t > ().valor || (int) yystack_[0].value.as < obj_t > ().valor;}
#line 1154 "../GenesysParser.cpp"
    break;

  case 25: // logica: expression oNAND expression
#line 270 "bisonparser.yy"
                                   { yylhs.value.as < obj_t > ().valor = !((int) yystack_[2].value.as < obj_t > ().valor && (int) yystack_[0].value.as < obj_t > ().valor);}
#line 1160 "../GenesysParser.cpp"
    break;

  case 26: // logica: expression oXOR expression
#line 271 "bisonparser.yy"
                                   { yylhs.value.as < obj_t > ().valor = (!(int) yystack_[2].value.as < obj_t > ().valor && (int) yystack_[0].value.as < obj_t > ().valor) || ((int) yystack_[2].value.as < obj_t > ().valor && !(int) yystack_[0].value.as < obj_t > ().valor);}
#line 1166 "../GenesysParser.cpp"
    break;

  case 27: // logica: oNOT expression
#line 272 "bisonparser.yy"
                                   { yylhs.value.as < obj_t > ().valor = !(int) yystack_[0].value.as < obj_t > ().valor;}
#line 1172 "../GenesysParser.cpp"
    break;

  case 28: // relacional: expression "<" expression
#line 276 "bisonparser.yy"
                                       { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor < yystack_[0].value.as < obj_t > ().valor ? 1 : 0;}
#line 1178 "../GenesysParser.cpp"
    break;

  case 29: // relacional: expression ">" expression
#line 277 "bisonparser.yy"
                                       { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor > yystack_[0].value.as < obj_t > ().valor ? 1 : 0;}
#line 1184 "../GenesysParser.cpp"
    break;

  case 30: // relacional: expression oLE expression
#line 278 "bisonparser.yy"
                                       { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor <= yystack_[0].value.as < obj_t > ().valor ? 1 : 0;}
#line 1190 "../GenesysParser.cpp"
    break;

  case 31: // relacional: expression oGE expression
#line 279 "bisonparser.yy"
                                       { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as <
```

```
obj_t > ().valor >= yystack_[0].value.as < obj_t > ().valor ? 1 : 0;}
#line 1196 "../GenesysParser.cpp"
    break;

  case 32: // relacional: expression oEQ expression
#line 280 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor == yystack_[0].value.as < obj_t > ().valor ? 1 : 0;}
#line 1202 "../GenesysParser.cpp"
    break;

  case 33: // relacional: expression oNE expression
#line 281 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = yystack_[2].value.as < obj_t > ().valor != yystack_[0].value.as < obj_t > ().valor ? 1 : 0;}
#line 1208 "../GenesysParser.cpp"
    break;

  case 34: // comando: comandoIF
#line 285 "bisonparser.yy"
                    { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1214 "../GenesysParser.cpp"
    break;

  case 35: // comando: comandoFOR
#line 286 "bisonparser.yy"
                    { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1220 "../GenesysParser.cpp"
    break;

  case 36: // comandoIF: cIF expression expression cELSE expression
#line 290 "bisonparser.yy"
                                                { yylhs.value.as < obj_t > ().valor = yystack_[3].value.as < obj_t > ().valor != 0 ? yystack_[2].value.as < obj_t > ().valor : yystack_[0].value.as < obj_t > ().valor; }
#line 1226 "../GenesysParser.cpp"
    break;

  case 37: // comandoIF: cIF expression expression
#line 291 "bisonparser.yy"
                                               { yylhs.value.as < obj_t > ().valor = yystack_[1].value.as < obj_t > ().valor != 0 ? yystack_[0].value.as < obj_t > ().valor : 0;}
#line 1232 "../GenesysParser.cpp"
    break;

  case 38: // comandoFOR: cFOR variavel "=" expression cTO expression cDO atribuicao
#line 296 "bisonparser.yy"
                                                                  {yylhs.value.as < obj_t > ().valor = 0; }
#line 1238 "../GenesysParser.cpp"
    break;

  case 39: // comandoFOR: cFOR atributo "=" expression cTO expression cDO atribuicao
#line 297 "bisonparser.yy"
                                                                   {yylhs.value.as < obj_t > ().valor = 0; }
#line 1244 "../GenesysParser.cpp"
    break;

  case 40: // function: mathFunction
#line 301 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1250 "../GenesysParser.cpp"
    break;

  case 41: // function: trigonFunction
#line 302 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1256 "../GenesysParser.cpp"
    break;

  case 42: // function: probFunction
#line 303 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1262 "../GenesysParser.cpp"
    break;

  case 43: // function: kernelFunction
#line 304 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1268 "../GenesysParser.cpp"
    break;
```

```
  case 44: // function: pluginFunction
#line 305 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1274 "../GenesysParser.cpp"
    break;

  case 45: // function: userFunction
#line 306 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1280 "../GenesysParser.cpp"
    break;

  case 46: // kernelFunction: fTNOW
#line 310 "bisonparser.yy"
                { yylhs.value.as < obj_t > ().valor = driver.getModel()->getSimulation()->getSimulatedTime();}
#line 1286 "../GenesysParser.cpp"
    break;

  case 47: // kernelFunction: fTFIN
#line 311 "bisonparser.yy"
                { yylhs.value.as < obj_t > ().valor = driver.getModel()->getSimulation()->getReplicationLength();}
#line 1292 "../GenesysParser.cpp"
    break;

  case 48: // kernelFunction: CSTAT
#line 312 "bisonparser.yy"
                        { yylhs.value.as < obj_t > ().valor = 0; }
#line 1298 "../GenesysParser.cpp"
    break;

  case 49: // kernelFunction: fTAVG "(" CSTAT ")"
#line 313 "bisonparser.yy"
                                {
                    StatisticsCollector* cstat = ((StatisticsCollector*)(driver.getModel()->getElements()->getElement(Util::TypeOf<StatisticsCollector>(), yystack_[1].value.as < obj_t > ().id)));
                    double value = cstat->getStatistics()->average();
                    yylhs.value.as < obj_t > ().valor = value; }
#line 1307 "../GenesysParser.cpp"
    break;

  case 50: // trigonFunction: fSIN "(" expression ")"
#line 320 "bisonparser.yy"
                                { yylhs.value.as < obj_t > ().valor = sin(yystack_[1].value.as < obj_t > ().valor); }
#line 1313 "../GenesysParser.cpp"
    break;

  case 51: // trigonFunction: fCOS "(" expression ")"
#line 321 "bisonparser.yy"
                                { yylhs.value.as < obj_t > ().valor = cos(yystack_[1].value.as < obj_t > ().valor); }
#line 1319 "../GenesysParser.cpp"
    break;

  case 52: // mathFunction: fROUND "(" expression ")"
#line 325 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = round(yystack_[1].value.as < obj_t > ().valor);}
#line 1325 "../GenesysParser.cpp"
    break;

  case 53: // mathFunction: fFRAC "(" expression ")"
#line 326 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = yystack_[1].value.as < obj_t > ().valor - (int) yystack_[1].value.as < obj_t > ().valor;}
#line 1331 "../GenesysParser.cpp"
    break;

  case 54: // mathFunction: fTRUNC "(" expression ")"
#line 327 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = trunc(yystack_[1].value.as < obj_t > ().valor);}
#line 1337 "../GenesysParser.cpp"
    break;

  case 55: // mathFunction: fEXP "(" expression ")"
#line 328 "bisonparser.yy"
                                  { yylhs.value.as < obj_t > ().valor = exp(yystack_[1].value.as < obj_t > ().valor);}
#line 1343 "../GenesysParser.cpp"
    break;

```

```
  case 56: // mathFunction: fSQRT "(" expression ")"
#line 329 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = sqrt(yystack_[1].value.as < obj_t > ().valor);}
#line 1349 "../GenesysParser.cpp"
    break;

  case 57: // mathFunction: fLOG "(" expression ")"
#line 330 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = log10(yystack_[1].value.as < obj_t > ().valor);}
#line 1355 "../GenesysParser.cpp"
    break;

  case 58: // mathFunction: fLN "(" expression ")"
#line 331 "bisonparser.yy"
                                    { yylhs.value.as < obj_t > ().valor = log(yystack_[1].value.as < obj_t > ().valor);}
#line 1361 "../GenesysParser.cpp"
    break;

  case 59: // mathFunction: fMOD "(" expression "," expression ")"
#line 332 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = (int) yystack_[3].value.as < obj_t > ().valor % (int) yystack_[1].value.as < obj_t > ().valor; }
#line 1367 "../GenesysParser.cpp"
    break;

  case 60: // probFunction: fRND1
#line 336 "bisonparser.yy"
                                                { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleUniform(0.0,1.0);}
#line 1373 "../GenesysParser.cpp"
    break;

  case 61: // probFunction: fEXPO "(" expression ")"
#line 337 "bisonparser.yy"
                                     { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleExponential(yystack_[1].value.as < obj_t > ().valor);}
#line 1379 "../GenesysParser.cpp"
    break;

  case 62: // probFunction: fNORM "(" expression "," expression ")"
#line 338 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleNormal(yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1385 "../GenesysParser.cpp"
    break;

  case 63: // probFunction: fUNIF "(" expression "," expression ")"
#line 339 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleUniform(yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1391 "../GenesysParser.cpp"
    break;

  case 64: // probFunction: fWEIB "(" expression "," expression ")"
#line 340 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleWeibull(yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1397 "../GenesysParser.cpp"
    break;

  case 65: // probFunction: fLOGN "(" expression "," expression ")"
#line 341 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleLogNormal(yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1403 "../GenesysParser.cpp"
    break;

  case 66: // probFunction: fGAMM "(" expression "," expression ")"
#line 342 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleGamma(yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1409 "../GenesysParser.cpp"
    break;

  case 67: // probFunction: fERLA "(" expression "," expression ")"
#line 343 "bisonparser.yy"
                                            { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleErlang(yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
```

```
#line 1415 "../GenesysParser.cpp"
    break;

  case 68: // probFunction: fTRIA "(" expression "," expression "," expression ")"
#line 344 "bisonparser.yy"
                                                                { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleTriangular(yystack_[5].value.as < obj_t > ().valor,yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1421 "../GenesysParser.cpp"
    break;

  case 69: // probFunction: fBETA "(" expression "," expression "," expression "," expression ")"
#line 345 "bisonparser.yy"
                                                                                { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleBeta(yystack_[7].value.as < obj_t > ().valor,yystack_[5].value.as < obj_t > ().valor,yystack_[3].value.as < obj_t > ().valor,yystack_[1].value.as < obj_t > ().valor);}
#line 1427 "../GenesysParser.cpp"
    break;

  case 70: // probFunction: fDISC "(" listaparm ")"
#line 346 "bisonparser.yy"
                                                { yylhs.value.as < obj_t > ().valor = driver.sampler()->sampleDiscrete(0,0); /*@TODO: NOT IMPLEMENTED YET*/ }
#line 1433 "../GenesysParser.cpp"
    break;

  case 71: // userFunction: "USER" "(" expression ")"
#line 352 "bisonparser.yy"
                                                  { yylhs.value.as < obj_t > ().valor = yystack_[1].value.as < obj_t > ().valor; }
#line 1439 "../GenesysParser.cpp"
    break;

  case 72: // listaparm: listaparm "," expression "," expression
#line 357 "bisonparser.yy"
                                                    {/*@TODO: NOT IMPLEMENTED YET*/}
#line 1445 "../GenesysParser.cpp"
    break;

  case 73: // listaparm: expression "," expression
#line 358 "bisonparser.yy"
                                                    {/*@TODO: NOT IMPLEMENTED YET*/}
#line 1451 "../GenesysParser.cpp"
    break;

  case 74: // atributo: ATRIB
#line 384 "bisonparser.yy"
                    {
        double attributeValue = 0.0;
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(yystack_[0].value.as < obj_t > ().id);
        }
        //std::cout « "Passei" « std::endl;
        yylhs.value.as < obj_t > ().valor = attributeValue;
    }
#line 1465 "../GenesysParser.cpp"
    break;

  case 75: // atributo: ATRIB "[" expression "]"
#line 393 "bisonparser.yy"
                                                {
        double attributeValue = 0.0;
        std::string index = std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(index, yystack_[3].value.as < obj_t > ().id);
        }
        yylhs.value.as < obj_t > ().valor = attributeValue;
    }
#line 1479 "../GenesysParser.cpp"
    break;

  case 76: // atributo: ATRIB "[" expression "," expression "]"
#line 402 "bisonparser.yy"
                                                            {
        double attributeValue = 0.0;
        std::string index = std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
```

```
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(index, yystack_[5].value.as < obj_t > ().id);
        }
        yylhs.value.as < obj_t > ().valor = attributeValue;
    }
#line 1493 "../GenesysParser.cpp"
    break;

  case 77: // atributo: ATRIB "[" expression "," expression "," expression "]"
#line 411 "bisonparser.yy"
                                                                    {
        double attributeValue = 0.0;
        std::string index = std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(index, yystack_[7].value.as < obj_t > ().id);
        }
        yylhs.value.as < obj_t > ().valor = attributeValue;
    }
#line 1507 "../GenesysParser.cpp"
    break;

  case 78: // variavel: VARI
#line 425 "bisonparser.yy"
                                {yylhs.value.as < obj_t > ().valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[0].value.as < obj_t > ().id)))->value();}
#line 1513 "../GenesysParser.cpp"
    break;

  case 79: // variavel: VARI "[" expression "]"
#line 426 "bisonparser.yy"
                                                                    {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
                    yylhs.value.as < obj_t > ().valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[3].value.as < obj_t > ().id)))->value(index); }
#line 1521 "../GenesysParser.cpp"
    break;

  case 80: // variavel: VARI "[" expression "," expression "]"
#line 429 "bisonparser.yy"
                                                                                        {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
                    yylhs.value.as < obj_t > ().valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[5].value.as < obj_t > ().id)))->value(index);}
#line 1529 "../GenesysParser.cpp"
    break;

  case 81: // variavel: VARI "[" expression "," expression "," expression "]"
#line 432 "bisonparser.yy"
                                                                                                                {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
                    yylhs.value.as < obj_t > ().valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[7].value.as < obj_t > ().id)))->value(index);}
#line 1537 "../GenesysParser.cpp"
    break;

  case 82: // formula: FORM
#line 440 "bisonparser.yy"
                                    {
                    yylhs.value.as < obj_t > ().valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), yystack_[0].value.as < obj_t > ().id)))->value();}
#line 1544 "../GenesysParser.cpp"
    break;

  case 83: // formula: FORM "[" expression "]"
#line 442 "bisonparser.yy"
                                                                {
```

```
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
                    yylhs.value.as < obj_t > ().valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), yystack_[3].value.as < obj_t > ().id)))->value(index);}
#line 1552 "../GenesysParser.cpp"
    break;

  case 84: // formula: FORM "[" expression "," expression "]"
#line 445 "bisonparser.yy"
                                                                                                    {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor)) +","+std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
                    yylhs.value.as < obj_t > ().valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), yystack_[5].value.as < obj_t > ().id)))->value(index);}
#line 1560 "../GenesysParser.cpp"
    break;

  case 85: // formula: FORM "[" expression "," expression "," expression "]"
#line 448 "bisonparser.yy"
                                                                                                                                {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor)) +","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[1].value.as < obj_t > ().valor));
                    yylhs.value.as < obj_t > ().valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), yystack_[7].value.as < obj_t > ().id)))->value(index);}
#line 1568 "../GenesysParser.cpp"
    break;

  case 86: // atribuicao: ATRIB "=" expression
#line 456 "bisonparser.yy"
                                                    {
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue("", yystack_[2].value.as < obj_t > ().id, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1576 "../GenesysParser.cpp"
    break;

  case 87: // atribuicao: ATRIB "[" expression "]" "=" expression
#line 459 "bisonparser.yy"
                                                                                                        {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor));
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue(index, yystack_[5].value.as < obj_t > ().id, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1585 "../GenesysParser.cpp"
    break;

  case 88: // atribuicao: ATRIB "[" expression "," expression "]" "=" expression
#line 463 "bisonparser.yy"

    {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor));
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue(index, yystack_[7].value.as < obj_t > ().id, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor;}
#line 1594 "../GenesysParser.cpp"
    break;

  case 89: // atribuicao: ATRIB "[" expression "," expression "," expression "]" "=" expression
#line 467 "bisonparser.yy"

                    {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[7].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor));
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue(index, yystack_[9].value.as < obj_t > ().id, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1603 "../GenesysParser.cpp"
    break;

  case 90: // atribuicao: VARI "=" expression
#line 473 "bisonparser.yy"
                                                                    {

    ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[2].value.as < obj_t > ().id)))->setValue(yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1611 "../GenesysParser.cpp"
    break;
```

```
  case 91: // atribuicao: VARI "[" expression "]" "=" expression
#line 476 "bisonparser.yy"
                                                                                                    {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor));
 ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[5].value.as < obj_t > ().id)))->setValue(index, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1620 "../GenesysParser.cpp"
    break;

  case 92: // atribuicao: VARI "[" expression "," expression "]" "=" expression
#line 480 "bisonparser.yy"

 {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor));

 ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[7].value.as < obj_t > ().id)))->setValue(index, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1629 "../GenesysParser.cpp"
    break;

  case 93: // atribuicao: VARI "[" expression "," expression "," expression "]" "=" expression
#line 484 "bisonparser.yy"

                    {
                    std::string index = std::to_string(static_cast<unsigned int>(yystack_[7].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[5].value.as < obj_t > ().valor))+","+std::to_string(static_cast<unsigned int>(yystack_[3].value.as < obj_t > ().valor));

 ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), yystack_[9].value.as < obj_t > ().id)))->setValue(index, yystack_[0].value.as < obj_t > ().valor);
                    yylhs.value.as < obj_t > ().valor = yystack_[0].value.as < obj_t > ().valor; }
#line 1638 "../GenesysParser.cpp"
    break;

  case 94: // pluginFunction: CTEZERO
#line 495 "bisonparser.yy"
                                                        { yylhs.value.as < obj_t > ().valor = 0; }
#line 1644 "../GenesysParser.cpp"
    break;

  case 95: // pluginFunction: fNQ "(" QUEUE ")"
#line 498 "bisonparser.yy"
                                    {   //std::cout « "Queue ID: " « $3.id « ", Size: " « ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->size() « std::endl;
                                        yylhs.value.as < obj_t > ().valor = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), yystack_[1].value.as < obj_t > ().id)))->size();}
#line 1651 "../GenesysParser.cpp"
    break;

  case 96: // pluginFunction: fLASTINQ "(" QUEUE ")"
#line 500 "bisonparser.yy"
                                    {/*For now does nothing because need acces to list of QUEUE, or at least the last element*/ }
#line 1657 "../GenesysParser.cpp"
    break;

  case 97: // pluginFunction: fFIRSTINQ "(" QUEUE ")"
#line 501 "bisonparser.yy"
                                    {
                if (((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), yystack_[1].value.as < obj_t > ().id)))->size() > 0){
                    //id da 1a entidade da fila, talvez pegar nome
                    yylhs.value.as < obj_t > ().valor = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), yystack_[1].value.as < obj_t > ().id)))->first()->getEntity()->getId();
                }else{
                    yylhs.value.as < obj_t > ().valor = 0;
                } }
#line 1669 "../GenesysParser.cpp"
    break;

  case 98: // pluginFunction: fSAQUE "(" QUEUE "," ATRIB ")"
#line 508 "bisonparser.yy"
                                        {
                Util::identification queueID = yystack_[3].value.as < obj_t > ().id;
                Util::identification attrID = yystack_[1].value.as < obj_t > ().id;
                double sum = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), yystack_[3].value.as <
```

```
obj_t > ().id)))->sumAttributesFromWaiting(attrID);
                yylhs.value.as < obj_t > ().valor = sum; }
#line 1679 "../GenesysParser.cpp"
    break;

  case 99: // pluginFunction: fAQUE "(" QUEUE "," NUMD "," ATRIB ")"
#line 513 "bisonparser.yy"
                                                {
                Util::identification queueID = yystack_[5].value.as < obj_t > ().id;
                Util::identification attrID = yystack_[1].value.as < obj_t > ().id;
                double value =
((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), yystack_[5].value.as < obj_t > ().id)))->getAttributeFromWaitingRank(yystack_[3].value.as < obj_t > ().valor-1, attrID); // rank starts on 0 in genesys
                yylhs.value.as < obj_t > ().valor = value; }
#line 1689 "../GenesysParser.cpp"
    break;

  case 100: // pluginFunction: fMR "(" RESOURCE ")"
#line 521 "bisonparser.yy"
                                        { yylhs.value.as < obj_t > ().valor =
((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), yystack_[1].value.as < obj_t > ().id))->getCapacity();}
#line 1695 "../GenesysParser.cpp"
    break;

  case 101: // pluginFunction: fNR "(" RESOURCE ")"
#line 522 "bisonparser.yy"
                                        { yylhs.value.as < obj_t > ().valor =
((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), yystack_[1].value.as < obj_t > ().id))->getNumberBusy();}
#line 1701 "../GenesysParser.cpp"
    break;

  case 102: // pluginFunction: fRESSEIZES "(" RESOURCE ")"
#line 523 "bisonparser.yy"
                                        { /*For now does nothing because needs get Seizes, check with teacher*/}
#line 1707 "../GenesysParser.cpp"
    break;

  case 103: // pluginFunction: fSTATE "(" RESOURCE ")"
#line 524 "bisonparser.yy"
                                        {  yylhs.value.as < obj_t > ().valor =
static_cast<int>(((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), yystack_[1].value.as < obj_t > ().id))->getResourceState()); }
#line 1713 "../GenesysParser.cpp"
    break;

  case 104: // pluginFunction: fIRF "(" RESOURCE ")"
#line 525 "bisonparser.yy"
                                        { yylhs.value.as < obj_t > ().valor =
((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), yystack_[1].value.as < obj_t > ().id))->getResourceState() == Resource::ResourceState::FAILED ? 1 : 0; }
#line 1719 "../GenesysParser.cpp"
    break;

  case 105: // pluginFunction: fSETSUM "(" SET ")"
#line 526 "bisonparser.yy"
                                {
                unsigned int count=0;
                Resource* res;
                List<ModelElement*>* setList =
((Set*)driver.getModel()->getElements()->getElement(Util::TypeOf<Set>(),yystack_[1].value.as < obj_t > ().id))->getElementSet();
                for (std::list<ModelElement*>::iterator it = setList->list()->begin(); it!=setList->list()->end(); it++) {
                    res = dynamic_cast<Resource*>(*it);
                    if (res != nullptr) {
                        if (res->getResourceState()==Resource::ResourceState::BUSY) {
                            count++;
                        }
                    }
                }
                yylhs.value.as < obj_t > ().valor = count; }
#line 1737 "../GenesysParser.cpp"
    break;

  case 106: // pluginFunction: fNUMSET "(" SET ")"
#line 542 "bisonparser.yy"
                                { yylhs.value.as < obj_t > ().valor =
((Set*)driver.getModel()->getElements()->getElement(Util::TypeOf<Set>(),yystack_[1].value.as < obj_t > ().id))->getElementSet()->size(); }
#line 1743 "../GenesysParser.cpp"
    break;
```

```

#line 1747 "../GenesysParser.cpp"

            default:
              break;
            }
        }
#if YY_EXCEPTIONS
      catch (const syntax_error& yyexc)
        {
          YYCDEBUG « "Caught exception: " « yyexc.what() « '\n';
          error (yyexc);
          YYERROR;
        }
#endif // YY_EXCEPTIONS
      YY_SYMBOL_PRINT ("-> $$ =", yylhs);
      yypop_ (yylen);
      yylen = 0;

      // Shift the result of the reduction.
      yypush_ (YY_NULLPTR, YY_MOVE (yylhs));
    }
    goto yynewstate;


  /*--------------------------------------.
  | yyerrlab -- here on detecting error.  |
  `--------------------------------------*/
  yyerrlab:
    // If not already recovering from an error, report this error.
    if (!yyerrstatus_)
      {
        ++yynerrs_;
        context yyctx (*this, yyla);
        std::string msg = yysyntax_error_ (yyctx);
        error (yyla.location, YY_MOVE (msg));
      }


    yyerror_range[1].location = yyla.location;
    if (yyerrstatus_ == 3)
      {
        /* If just tried and failed to reuse lookahead token after an
           error, discard it.  */

        // Return failure if at end of input.
        if (yyla.kind () == symbol_kind::S_YYEOF)
          YYABORT;
        else if (!yyla.empty ())
          {
            yy_destroy_ ("Error: discarding", yyla);
            yyla.clear ();
          }
      }

    // Else will try to reuse lookahead token after shifting the error token.
    goto yyerrlab1;


  /*---------------------------------------------------.
  | yyerrorlab -- error raised explicitly by YYERROR.  |
  `---------------------------------------------------*/
  yyerrorlab:
    /* Pacify compilers when the user code never invokes YYERROR and
       the label yyerrorlab therefore never appears in user code.  */
    if (false)
      YYERROR;

    /* Do not reclaim the symbols of the rule whose action triggered
       this YYERROR.  */
    yypop_ (yylen);
    yylen = 0;
    YY_STACK_PRINT ();
    goto yyerrlab1;


  /*-------------------------------------------------------------.
  | yyerrlab1 -- common code for both syntax error and YYERROR.  |
  `-------------------------------------------------------------*/
  yyerrlab1:
    yyerrstatus_ = 3;   // Each real token shifted decrements this.
    // Pop stack until we find a state that shifts the error token.
    for (;;)
      {
        yyn = yypact_[+yystack_[0].state];
        if (!yy_pact_value_is_default_ (yyn))
          {
```

```
            yyn += symbol_kind::S_YYerror;
            if (0 <= yyn && yyn <= yylast_
                && yycheck_[yyn] == symbol_kind::S_YYerror)
              {
                yyn = yytable_[yyn];
                if (0 < yyn)
                  break;
              }
          }

        // Pop the current state because it cannot handle the error token.
        if (yystack_.size () == 1)
          YYABORT;

        yyerror_range[1].location = yystack_[0].location;
        yy_destroy_ ("Error: popping", yystack_[0]);
        yypop_ ();
        YY_STACK_PRINT ();
      }
    {
      stack_symbol_type error_token;

      yyerror_range[2].location = yyla.location;
      YYLLOC_DEFAULT (error_token.location, yyerror_range, 2);

      // Shift the error token.
      error_token.state = state_type (yyn);
      yypush_ ("Shifting", YY_MOVE (error_token));
    }
    goto yynewstate;


  /*-------------------------------------.
  | yyacceptlab -- YYACCEPT comes here.  |
  `-------------------------------------*/
  yyacceptlab:
    yyresult = 0;
    goto yyreturn;


  /*-----------------------------------.
  | yyabortlab -- YYABORT comes here.  |
  `-----------------------------------*/
  yyabortlab:
    yyresult = 1;
    goto yyreturn;


  /*-----------------------------------------------------.
  | yyreturn -- parsing is finished, return the result.  |
  `-----------------------------------------------------*/
  yyreturn:
    if (!yyla.empty ())
      yy_destroy_ ("Cleanup: discarding lookahead", yyla);

    /* Do not reclaim the symbols of the rule whose action triggered
       this YYABORT or YYACCEPT.  */
    yypop_ (yylen);
    YY_STACK_PRINT ();
    while (1 < yystack_.size ())
      {
        yy_destroy_ ("Cleanup: popping", yystack_[0]);
        yypop_ ();
      }

    return yyresult;
  }
#if YY_EXCEPTIONS
    catch (...)
      {
        YYCDEBUG « "Exception caught: cleaning lookahead and stack\n";
        // Do not try to display the values of the reclaimed symbols,
        // as their printers might throw an exception.
        if (!yyla.empty ())
          yy_destroy_ (YY_NULLPTR, yyla);

        while (1 < yystack_.size ())
          {
            yy_destroy_ (YY_NULLPTR, yystack_[0]);
            yypop_ ();
          }
        throw;
      }
#endif // YY_EXCEPTIONS
  }

  void
```

```
genesyspp_parser::error (const syntax_error& yyexc)
{
  error (yyexc.location, yyexc.what ());
}

/* Return YYSTR after stripping away unnecessary quotes and
   backslashes, so that it's suitable for yyerror.  The heuristic is
   that double-quoting is unnecessary unless the string contains an
   apostrophe, a comma, or backslash (other than backslash-backslash).
   YYSTR is taken from yytname.  */
std::string
genesyspp_parser::yytnamerr_ (const char *yystr)
{
  if (*yystr == '"')
    {
      std::string yyr;
      char const *yyp = yystr;

      for (;;)
        switch (*++yyp)
          {
          case '\'':
          case ',':
            goto do_not_strip_quotes;

          case '\\':
            if (*++yyp != '\\')
              goto do_not_strip_quotes;
            else
              goto append;

          append:
          default:
            yyr += *yyp;
            break;

          case '"':
            return yyr;
          }
    do_not_strip_quotes: ;
    }

  return yystr;
}

std::string
genesyspp_parser::symbol_name (symbol_kind_type yysymbol)
{
  return yytnamerr_ (yytname_[yysymbol]);
}



// genesyspp_parser::context.
genesyspp_parser::context::context (const genesyspp_parser& yyparser, const symbol_type& yyla)
  : yyparser_ (yyparser)
  , yyla_ (yyla)
{}

int
genesyspp_parser::context::expected_tokens (symbol_kind_type yyarg[], int yyargn) const
{
  // Actual number of expected tokens
  int yycount = 0;

  int yyn = yypact_[+yyparser_.yystack_[0].state];
  if (!yy_pact_value_is_default_ (yyn))
    {
      /* Start YYX at -YYN if negative to avoid negative indexes in
         YYCHECK.  In other words, skip the first -YYN actions for
         this state because they are default actions.  */
      int yyxbegin = yyn < 0 ? -yyn : 0;
      // Stay within bounds of both yycheck and yytname.
      int yychecklim = yylast_ - yyn + 1;
      int yyxend = yychecklim < YYNTOKENS ? yychecklim : YYNTOKENS;
      for (int yyx = yyxbegin; yyx < yyxend; ++yyx)
        if (yycheck_[yyx + yyn] == yyx && yyx != symbol_kind::S_YYerror
            && !yy_table_value_is_error_ (yytable_[yyx + yyn]))
          {
            if (!yyarg)
              ++yycount;
            else if (yycount == yyargn)
              return 0;
            else
              yyarg[yycount++] = YY_CAST (symbol_kind_type, yyx);
          }
    }
```

```
  if (yyarg && yycount == 0 && 0 < yyargn)
    yyarg[0] = symbol_kind::S_YYEMPTY;
  return yycount;
}



int
genesyspp_parser::yy_syntax_error_arguments_ (const context& yyctx,
                                              symbol_kind_type yyarg[], int yyargn) const
{
  /* There are many possibilities here to consider:
     - If this state is a consistent state with a default action, then
       the only way this function was invoked is if the default action
       is an error action.  In that case, don't check for expected
       tokens because there are none.
     - The only way there can be no lookahead present (in yyla) is
       if this state is a consistent state with a default action.
       Thus, detecting the absence of a lookahead is sufficient to
       determine that there is no unexpected or expected token to
       report.  In that case, just report a simple "syntax error".
     - Don't assume there isn't a lookahead just because this state is
       a consistent state with a default action.  There might have
       been a previous inconsistent state, consistent state with a
       non-default action, or user semantic action that manipulated
       yyla.  (However, yyla is currently not documented for users.)
     - Of course, the expected token list depends on states to have
       correct lookahead information, and it depends on the parser not
       to perform extra reductions after fetching a lookahead from the
       scanner and before detecting a syntax error.  Thus, state merging
       (from LALR or IELR) and default reductions corrupt the expected
       token list.  However, the list is correct for canonical LR with
       one exception: it will still contain any token that will not be
       accepted due to an error action in a later state.
  */

  if (!yyctx.lookahead ().empty ())
    {
      if (yyarg)
        yyarg[0] = yyctx.token ();
      int yyn = yyctx.expected_tokens (yyarg ? yyarg + 1 : yyarg, yyargn - 1);
      return yyn + 1;
    }
  return 0;
}

// Generate an error message.
std::string
genesyspp_parser::yysyntax_error_ (const context& yyctx) const
{
  // Its maximum.
  enum { YYARGS_MAX = 5 };
  // Arguments of yyformat.
  symbol_kind_type yyarg[YYARGS_MAX];
  int yycount = yy_syntax_error_arguments_ (yyctx, yyarg, YYARGS_MAX);

  char const* yyformat = YY_NULLPTR;
  switch (yycount)
    {
#define YYCASE_(N, S)                         \
      case N:                                 \
        yyformat = S;                         \
      break
    default: // Avoid compiler warnings.
      YYCASE_ (0, YY_("syntax error"));
      YYCASE_ (1, YY_("syntax error, unexpected %s"));
      YYCASE_ (2, YY_("syntax error, unexpected %s, expecting %s"));
      YYCASE_ (3, YY_("syntax error, unexpected %s, expecting %s or %s"));
      YYCASE_ (4, YY_("syntax error, unexpected %s, expecting %s or %s or %s"));
      YYCASE_ (5, YY_("syntax error, unexpected %s, expecting %s or %s or %s or %s"));
#undef YYCASE_
    }

  std::string yyres;
  // Argument number.
  std::ptrdiff_t yyi = 0;
  for (char const* yyp = yyformat; *yyp; ++yyp)
    if (yyp[0] == '%' && yyp[1] == 's' && yyi < yycount)
      {
        yyres += symbol_name (yyarg[yyi++]);
        ++yyp;
      }
    else
      yyres += *yyp;
  return yyres;
}
```

```
const short genesyspp_parser::yypact_ninf_ = -298;

const signed char genesyspp_parser::yytable_ninf_ = -23;

const short
genesyspp_parser::yypact_[] =
{
   331,  -82, -298, -298, -298,  500,  -63,  -62,  -61,  -49,
   -45,  -42,  -41,  -40,  -39,  -36, -298,  -35,  -34,  -33,
   -32,  -29,  -28,  -27,  -26,  -25,  -24, -298, -298,  500,
   -48,  -69, -298,  -23,  -22,  -21,  -20,  -19,  -18,  -17,
   -16,  -15,  -13,  -10,   -9,   -8,  -68,  -50,  500,  500,
    -7,   91, 1692, -298, -298, -298, -298, -298, -298, -298,
  -298, -298, -298, -298, -298, -298, -298, -298, -298, -298,
  -298, -298,  185,  500,  500,  500,  500,  500,  500,  500,
   500,  500,  500,  500,  500,  500,  500,  500,  500,  500,
   500,  500,  500,  418,   31,   35,   27,   29,  500,  500,
    80,   78,   79,   81,   82,   83,   65,   76,   77,   84,
    85,   86,   71,  500,  500,  500, 1035,    8,  500, -298,
   500,  500,  500,  500,  500,  500,  500,  500,  500,  500,
   500,  500,  500,  500,  500, 1045, 1055, 1134,  764, 1144,
  1154, 1164, 1174, 1184, 1194, 1273,  773,  854,  863,  872,
   881,  890,  899,  908,  917,  -72,  500, 1281,  500,  500,
   500,  500,  151, 1692,   66,   67,   68,   69,   73,   74,
    75,   90,   92,  101,   54,   93,  102,  162, 1692,  574,
  -298, 1291,   24,   24,   24,   24,  185,  185, 1692, 1692,
    18,   18,    8, 1692,    8,   24,   24, -298, -298, -298,
   500, -298, -298, -298, -298, -298, -298, -298,  500,  500,
   500,  500,  500,  500,  500,  500,  500, -298,  500,   15,
   500,  584,  594, 1299, 1311,   95,  500, -298, -298, -298,
  -298, -298, -298, -298, -298, -298, -298,  103,  163, -298,
    96,  500, -298,  500, -298, 1321, 1331, 1410, 1420, 1430,
  1440, 1450,  998, 1007, 1692, 1016,   24, -298,  500, -298,
   500,  500,  500,  500,  604,  107,   97,  500,  614,  624,
  -298, -298, -298, -298, -298, -298, -298,  500,  500,  500,
   634,  715, 1458, 1468, 1692,   99,  500, -298,  129, 1692,
   100,  500, -298,  500, 1547, 1025, 1692, -298,  500, -298,
   500,  -47,  -47,  500, 1555,  111,  500, 1563, 1571, -298,
   500, 1579, 1587,  -67,  -66, -298, -298, 1692,  104, -298,
  1692,  105, -298, 1597, -298, -298,  500,  500,  500,  500,
  -298,  725,  735, 1692, 1692,   95,  500,   96,  500,  745,
   755,   99,  500,  100,  500, 1605, 1684,  104,  105
};

const signed char
genesyspp_parser::yydefact_[] =
{
     0,     0,    15,    16,    94,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,    60,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,    46,    47,     0,
     0,    74,    48,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,    78,    82,     0,     0,
     0,     0,     2,     4,     8,     9,    10,     6,    34,    35,
     5,    43,    41,    40,    42,    45,    12,    13,    14,     7,
    44,     3,    27,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,    74,    78,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,    22,     0,     1,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,    37,     0,     0,
     0,     0,     0,    86,     0,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,    90,     0,
    11,     0,    30,    31,    32,    33,    23,    24,    25,    26,
    17,    18,    20,    21,    19,    28,    29,    50,    51,    52,
     0,    54,    53,    55,    56,    57,    58,    61,     0,     0,
     0,     0,     0,     0,     0,     0,     0,    70,     0,    18,
     0,     0,     0,     0,     0,    75,     0,    49,   101,   100,
   104,   102,   103,   105,    95,    97,    96,     0,     0,   106,
    79,     0,    83,     0,    71,     0,     0,     0,     0,     0,
     0,     0,     0,     0,    73,     0,    36,    75,     0,    79,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
    59,    62,    63,    64,    65,    66,    67,     0,     0,     0,
     0,     0,     0,     0,    87,    76,     0,    98,     0,    91,
    80,     0,    84,     0,     0,     0,    72,    76,     0,    80,
     0,     0,     0,     0,     0,     0,     0,     0,     0,    68,
     0,     0,     0,     0,     0,    39,    38,    88,    77,    99,
    92,    81,    85,     0,    77,    81,     0,     0,     0,     0,
    69,     0,     0,    89,    93,     0,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,     0
};

const short
```

```
genesyspp_parser::yypgoto_[] =
{
  -298,  -298,    -5,  -298,  -298,  -298,  -298,  -298,  -298,  -298,
  -298,  -298,  -298,  -298,  -298,  -298,  -298,   157,   158,  -298,
  -297,  -298
};

const unsigned char
genesyspp_parser::yydefgoto_[] =
{
     0,    51,    52,    53,    54,    55,    56,    57,    58,    59,
    60,    61,    62,    63,    64,    65,   155,    66,    67,    68,
    69,    70
};

const short
genesyspp_parser::yytable_[] =
{
    72,    94,   313,   217,   315,   316,    71,    98,   113,   326,
   327,    73,    74,    75,   218,   -22,    99,   114,    99,   114,
   126,   127,    95,   314,    93,    76,   115,   126,   127,    77,
   126,   127,    78,    79,    80,    81,   126,   127,    82,    83,
    84,    85,    86,   116,   117,    87,    88,    89,    90,    91,
    92,   100,   101,   102,   103,   104,   105,   106,   107,   108,
   -22,   109,   -22,   -22,   110,   111,   112,   118,   135,   136,
   137,   138,   139,   140,   141,   142,   143,   144,   145,   146,
   147,   148,   149,   150,   151,   152,   153,   154,   157,   131,
   -22,   119,   -22,   162,   163,   -22,   131,   -22,   130,   131,
   132,   -22,   128,   129,   130,   131,   132,   158,   177,   178,
   179,   159,   160,   181,   161,   182,   183,   184,   185,   186,
   187,   188,   189,   190,   191,   192,   193,   194,   195,   196,
   164,   165,   166,   170,   167,   168,   169,   171,   172,   176,
   237,   227,   228,   229,   230,   173,   174,   175,   231,   232,
   233,   219,   265,   221,   222,   223,   224,   120,   121,   122,
   123,   124,   125,   126,   127,   234,   266,   235,   120,   121,
   122,   123,   124,   125,   126,   127,   236,   239,   305,   238,
   263,   267,   287,   288,   303,   306,   319,    96,    97,   328,
   329,   120,   121,   122,   123,   245,     0,   126,   127,     0,
     0,     0,     0,   246,   247,   248,   249,   250,   251,   252,
   253,   254,     0,   255,     0,   256,     0,     0,     0,     0,
     0,   264,     0,     0,     0,     0,     0,     0,   225,   128,
   129,   130,   131,   132,   133,   134,   268,   226,   269,   240,
   128,   129,   130,   131,   132,   133,   134,     0,   241,     0,
     0,     0,     0,   280,     0,   281,   282,   283,   284,     0,
     0,     0,   289,   128,   129,   130,   131,   132,   133,   134,
     0,     0,   294,   295,   296,     0,     0,     0,     0,     0,
     0,   304,     0,     0,     0,     0,   307,     0,   308,     0,
     0,     0,     0,   311,     0,   312,     0,     0,   317,     0,
     0,   320,     0,     0,     0,   323,     0,     0,     0,     0,
     0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
     0,   331,   332,   333,   334,     0,     0,     0,     0,     0,
     0,   339,     1,   340,     2,     3,     4,   345,     0,   346,
     0,     0,     0,     0,     0,     5,     6,     7,     8,     9,
    10,    11,    12,    13,    14,    15,     0,     0,     0,    16,
    17,    18,    19,    20,    21,    22,    23,    24,    25,    26,
    27,    28,     0,     0,     0,    29,     0,    30,     0,     0,
    31,    32,    33,     0,     0,    34,    35,    36,    37,    38,
    39,     0,     0,    40,    41,    42,    43,    44,     0,     0,
    45,    46,    47,     0,     0,    48,     0,     0,     0,     0,
    49,     0,     0,     0,     0,     0,     0,     0,     0,     0,
    50,     2,     3,     4,   120,   121,   122,   123,   124,   125,
   126,   127,     5,     6,     7,     8,     9,    10,    11,    12,
    13,    14,    15,     0,     0,     0,    16,    17,    18,    19,
    20,    21,    22,    23,    24,    25,    26,    27,    28,     0,
     0,     0,    29,     0,    30,     0,     0,    31,    32,    33,
     0,     0,    34,    35,    36,    37,    38,    39,     0,     0,
    40,    41,    42,    43,    44,     0,     0,    45,    46,    47,
     0,     0,    48,     0,     0,     0,   128,   156,   130,   131,
   132,   133,   134,     2,     3,     4,     0,    50,     0,     0,
     0,     0,     0,     0,     5,     6,     7,     8,     9,    10,
    11,    12,    13,    14,    15,     0,     0,     0,    16,    17,
    18,    19,    20,    21,    22,    23,    24,    25,    26,    27,
    28,     0,     0,     0,    29,     0,    30,     0,     0,    31,
    32,    33,     0,     0,    34,    35,    36,    37,    38,    39,
     0,     0,    40,    41,    42,    43,    44,     0,     0,    45,
    46,    47,     0,     0,    48,     0,     0,     0,     0,    49,
   120,   121,   122,   123,   124,   125,   126,   127,     0,    50,
   120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
   120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
   120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
   120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
   120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
   120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
     0,   242,   128,   129,   130,   131,   132,   133,   134,     0,
   243,   257,   128,   129,   130,   131,   132,   133,   134,     0,
   258,   259,   128,   129,   130,   131,   132,   133,   134,     0,
   260,   285,   128,   129,   130,   131,   132,   133,   134,     0,
```

```
02267      286,   290,   128,   129,   130,   131,   132,   133,   134,     0,
02268      291,   292,   128,   129,   130,   131,   132,   133,   134,     0,
02269      293,   297,   128,   129,   130,   131,   132,   133,   134,     0,
02270      298,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02271        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02272        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02273        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02274        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02275      120,   121,   122,   123,   124,   125,   126,   127,     0,   120,
02276      121,   122,   123,   124,   125,   126,   127,     0,     0,     0,
02277        0,     0,   299,   128,   129,   130,   131,   132,   133,   134,
02278        0,   300,   335,   128,   129,   130,   131,   132,   133,   134,
02279        0,   336,   337,   128,   129,   130,   131,   132,   133,   134,
02280        0,   338,   341,   128,   129,   130,   131,   132,   133,   134,
02281        0,   342,   343,   128,   129,   130,   131,   132,   133,   134,
02282        0,   344,   128,   129,   130,   131,   132,   133,   134,     0,
02283      200,   128,   129,   130,   131,   132,   133,   134,     0,   208,
02284      120,   121,   122,   123,   124,   125,   126,   127,     0,   120,
02285      121,   122,   123,   124,   125,   126,   127,     0,   120,   121,
02286      122,   123,   124,   125,   126,   127,     0,   120,   121,   122,
02287      123,   124,   125,   126,   127,     0,   120,   121,   122,   123,
02288      124,   125,   126,   127,     0,   120,   121,   122,   123,   124,
02289      125,   126,   127,     0,   120,   121,   122,   123,   124,   125,
02290      126,   127,     0,   120,   121,   122,   123,   124,   125,   126,
02291      127,     0,   128,   129,   130,   131,   132,   133,   134,     0,
02292      209,   128,   129,   130,   131,   132,   133,   134,     0,   210,
02293      128,   129,   130,   131,   132,   133,   134,     0,   211,   128,
02294      129,   130,   131,   132,   133,   134,     0,   212,   128,   129,
02295      130,   131,   132,   133,   134,     0,   213,   128,   129,   130,
02296      131,   132,   133,   134,     0,   214,   128,   129,   130,   131,
02297      132,   133,   134,     0,   215,   128,   129,   130,   131,   132,
02298      133,   134,     0,   216,   120,   121,   122,   123,   124,   125,
02299      126,   127,     0,   120,   121,   122,   123,   124,   125,   126,
02300      127,     0,   120,   121,   122,   123,   124,   125,   126,   127,
02301        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02302        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02303        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02304        0,   120,   121,   122,   123,   124,   125,   126,   127,     0,
02305        0,     0,     0,     0,     0,     0,   128,   129,   130,   131,
02306      132,   133,   134,     0,   277,   128,   129,   130,   131,   132,
02307      133,   134,     0,   278,   128,   129,   130,   131,   132,   133,
02308      134,     0,   279,   128,   129,   130,   131,   132,   133,   134,
02309      180,   310,     0,   128,   129,   130,   131,   132,   133,   134,
02310      197,     0,     0,   128,   129,   130,   131,   132,   133,   134,
02311      198,     0,     0,   128,   129,   130,   131,   132,   133,   134,
02312      120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
02313      120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
02314      120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
02315      120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
02316      120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
02317      120,   121,   122,   123,   124,   125,   126,   127,     0,     0,
02318      120,   121,   122,   123,   124,   125,   126,   127,     0,   199,
02319        0,     0,   128,   129,   130,   131,   132,   133,   134,   201,
02320        0,     0,   128,   129,   130,   131,   132,   133,   134,   202,
02321        0,     0,   128,   129,   130,   131,   132,   133,   134,   203,
02322        0,     0,   128,   129,   130,   131,   132,   133,   134,   204,
02323        0,     0,   128,   129,   130,   131,   132,   133,   134,   205,
02324        0,     0,   128,   129,   130,   131,   132,   133,   134,   206,
02325        0,     0,   128,   129,   130,   131,   132,   133,   134,   120,
02326      121,   122,   123,   124,   125,   126,   127,   120,   121,   122,
02327      123,   124,   125,   126,   127,     0,     0,   120,   121,   122,
02328      123,   124,   125,   126,   127,   120,   121,   122,   123,   124,
02329      125,   126,   127,     0,     0,     0,     0,   120,   121,   122,
02330      123,   124,   125,   126,   127,     0,   220,   120,   121,   122,
02331      123,   124,   125,   126,   127,     0,     0,   120,   121,   122,
02332      123,   124,   125,   126,   127,     0,   261,     0,   207,     0,
02333        0,   128,   129,   130,   131,   132,   133,   134,   262,   128,
02334      129,   130,   131,   132,   133,   134,   244,     0,     0,   128,
02335      129,   130,   131,   132,   133,   134,     0,   128,   129,   130,
02336      131,   132,   133,   134,     0,     0,     0,     0,     0,   128,
02337      129,   130,   131,   132,   133,   134,   270,     0,     0,   128,
02338      129,   130,   131,   132,   133,   134,   271,     0,     0,   128,
02339      129,   130,   131,   132,   133,   134,   120,   121,   122,   123,
02340      124,   125,   126,   127,     0,     0,   120,   121,   122,   123,
02341      124,   125,   126,   127,     0,     0,   120,   121,   122,   123,
02342      124,   125,   126,   127,     0,     0,   120,   121,   122,   123,
02343      124,   125,   126,   127,     0,     0,   120,   121,   122,   123,
02344      124,   125,   126,   127,   120,   121,   122,   123,   124,   125,
02345      126,   127,     0,     0,   120,   121,   122,   123,   124,   125,
02346      126,   127,     0,     0,     0,   272,     0,     0,   128,   129,
02347      130,   131,   132,   133,   134,   273,     0,     0,   128,   129,
02348      130,   131,   132,   133,   134,   274,   301,     0,   128,   129,
02349      130,   131,   132,   133,   134,   275,   302,     0,   128,   129,
02350      130,   131,   132,   133,   134,   276,     0,     0,   128,   129,
02351      130,   131,   132,   133,   134,     0,   128,   129,   130,   131,
02352      132,   133,   134,     0,     0,     0,   128,   129,   130,   131,
02353      132,   133,   134,   120,   121,   122,   123,   124,   125,   126,
```

```
    127,   120,   121,   122,   123,   124,   125,   126,   127,   120,
    121,   122,   123,   124,   125,   126,   127,   120,   121,   122,
    123,   124,   125,   126,   127,   120,   121,   122,   123,   124,
    125,   126,   127,   120,   121,   122,   123,   124,   125,   126,
    127,     0,     0,   120,   121,   122,   123,   124,   125,   126,
    127,   120,   121,   122,   123,   124,   125,   126,   127,     0,
      0,     0,   309,     0,     0,   128,   129,   130,   131,   132,
    133,   134,   318,   128,   129,   130,   131,   132,   133,   134,
    321,   128,   129,   130,   131,   132,   133,   134,   322,   128,
    129,   130,   131,   132,   133,   134,   324,   128,   129,   130,
    131,   132,   133,   134,   325,   128,   129,   130,   131,   132,
    133,   134,   330,     0,     0,   128,   129,   130,   131,   132,
    133,   134,   347,   128,   129,   130,   131,   132,   133,   134,
    120,   121,   122,   123,   124,   125,   126,   127,   120,   121,
    122,   123,   124,   125,   126,   127,     0,     0,     0,     0,
      0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
      0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
      0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
      0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
      0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
      0,   348,   128,   129,   130,   131,   132,   133,   134,     0,
    128,   129,   130,   131,   132,   133,   134
};

const short
genesyspp_parser::yycheck_[] =
{
      5,    49,    49,    75,   301,   302,    88,    76,    76,    76,
     76,    74,    74,    74,    86,     0,    85,    85,    85,    85,
     12,    13,    70,    70,    29,    74,    76,    12,    13,    74,
     12,    13,    74,    74,    74,    74,    12,    13,    74,    74,
     74,    74,    74,    48,    49,    74,    74,    74,    74,    74,
     74,    74,    74,    74,    74,    74,    74,    74,    74,    74,
     45,    74,    47,    48,    74,    74,    74,    74,    73,    74,
     75,    76,    77,    78,    79,    80,    81,    82,    83,    84,
     85,    86,    87,    88,    89,    90,    91,    92,    93,    81,
     75,     0,    77,    98,    99,    80,    81,    82,    80,    81,
     82,    86,    78,    79,    80,    81,    82,    76,   113,   114,
    115,    76,    85,   118,    85,   120,   121,   122,   123,   124,
    125,   126,   127,   128,   129,   130,   131,   132,   133,   134,
     50,    53,    53,    68,    53,    53,    53,    61,    61,    68,
     86,    75,    75,    75,    75,    61,    61,    61,    75,    75,
     75,   156,    49,   158,   159,   160,   161,     6,     7,     8,
      9,    10,    11,    12,    13,    75,     3,    75,     6,     7,
      8,     9,    10,    11,    12,    13,    75,    75,    49,    86,
     85,    85,    75,    86,    85,    85,    75,    30,    30,    85,
     85,     6,     7,     8,     9,   200,    -1,    12,    13,    -1,
     -1,    -1,    -1,   208,   209,   210,   211,   212,   213,   214,
    215,   216,    -1,   218,    -1,   220,    -1,    -1,    -1,    -1,
     -1,   226,    -1,    -1,    -1,    -1,    -1,    -1,    77,    78,
     79,    80,    81,    82,    83,    84,   241,    86,   243,    77,
     78,    79,    80,    81,    82,    83,    84,    -1,    86,    -1,
     -1,    -1,    -1,   258,    -1,   260,   261,   262,   263,    -1,
     -1,    -1,   267,    78,    79,    80,    81,    82,    83,    84,
     -1,    -1,   277,   278,   279,    -1,    -1,    -1,    -1,    -1,
     -1,   286,    -1,    -1,    -1,    -1,   291,    -1,   293,    -1,
     -1,    -1,    -1,   298,    -1,   300,    -1,    -1,   303,    -1,
     -1,   306,    -1,    -1,    -1,   310,    -1,    -1,    -1,    -1,
     -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
     -1,   326,   327,   328,   329,    -1,    -1,    -1,    -1,    -1,
     -1,   336,     1,   338,     3,     4,     5,   342,    -1,   344,
     -1,    -1,    -1,    -1,    -1,    14,    15,    16,    17,    18,
     19,    20,    21,    22,    23,    24,    -1,    -1,    -1,    28,
     29,    30,    31,    32,    33,    34,    35,    36,    37,    38,
     39,    40,    -1,    -1,    -1,    44,    -1,    46,    -1,    -1,
     49,    50,    51,    -1,    -1,    54,    55,    56,    57,    58,
     59,    -1,    -1,    62,    63,    64,    65,    66,    -1,    -1,
     69,    70,    71,    -1,    -1,    74,    -1,    -1,    -1,    -1,
     79,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
     89,     3,     4,     5,     6,     7,     8,     9,    10,    11,
     12,    13,    14,    15,    16,    17,    18,    19,    20,    21,
     22,    23,    24,    -1,    -1,    -1,    28,    29,    30,    31,
     32,    33,    34,    35,    36,    37,    38,    39,    40,    -1,
     -1,    -1,    44,    -1,    46,    -1,    -1,    49,    50,    51,
     -1,    -1,    54,    55,    56,    57,    58,    59,    -1,    -1,
     62,    63,    64,    65,    66,    -1,    -1,    69,    70,    71,
     -1,    -1,    74,    -1,    -1,    -1,    78,    79,    80,    81,
     82,    83,    84,     3,     4,     5,    -1,    89,    -1,    -1,
     -1,    -1,    -1,    -1,    14,    15,    16,    17,    18,    19,
     20,    21,    22,    23,    24,    -1,    -1,    -1,    28,    29,
     30,    31,    32,    33,    34,    35,    36,    37,    38,    39,
     40,    -1,    -1,    -1,    44,    -1,    46,    -1,    -1,    49,
     50,    51,    -1,    -1,    54,    55,    56,    57,    58,    59,
     -1,    -1,    62,    63,    64,    65,    66,    -1,    -1,    69,
     70,    71,    -1,    -1,    74,    -1,    -1,    -1,    -1,    79,
      6,     7,     8,     9,    10,    11,    12,    13,    -1,    89,
      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
```

```
02441      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02442      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02443      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02444      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02445      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02446     -1,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02447     86,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02448     86,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02449     86,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02450     86,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02451     86,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02452     86,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
02453     86,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02454     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02455     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02456     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02457     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02458      6,     7,     8,     9,    10,    11,    12,    13,    -1,     6,
02459      7,     8,     9,    10,    11,    12,    13,    -1,    -1,    -1,
02460     -1,    -1,    77,    78,    79,    80,    81,    82,    83,    84,
02461     -1,    86,    77,    78,    79,    80,    81,    82,    83,    84,
02462     -1,    86,    77,    78,    79,    80,    81,    82,    83,    84,
02463     -1,    86,    77,    78,    79,    80,    81,    82,    83,    84,
02464     -1,    86,    77,    78,    79,    80,    81,    82,    83,    84,
02465     -1,    86,    78,    79,    80,    81,    82,    83,    84,    -1,
02466     86,    78,    79,    80,    81,    82,    83,    84,    -1,    86,
02467      6,     7,     8,     9,    10,    11,    12,    13,    -1,     6,
02468      7,     8,     9,    10,    11,    12,    13,    -1,     6,     7,
02469      8,     9,    10,    11,    12,    13,    -1,     6,     7,     8,
02470      9,    10,    11,    12,    13,    -1,     6,     7,     8,     9,
02471     10,    11,    12,    13,    -1,     6,     7,     8,     9,    10,
02472     11,    12,    13,    -1,     6,     7,     8,     9,    10,    11,
02473     12,    13,    -1,     6,     7,     8,     9,    10,    11,    12,
02474     13,    -1,    78,    79,    80,    81,    82,    83,    84,    -1,
02475     86,    78,    79,    80,    81,    82,    83,    84,    -1,    86,
02476     78,    79,    80,    81,    82,    83,    84,    -1,    86,    78,
02477     79,    80,    81,    82,    83,    84,    -1,    86,    78,    79,
02478     80,    81,    82,    83,    84,    -1,    86,    78,    79,    80,
02479     81,    82,    83,    84,    -1,    86,    78,    79,    80,    81,
02480     82,    83,    84,    -1,    86,    78,    79,    80,    81,    82,
02481     83,    84,    -1,    86,     6,     7,     8,     9,    10,    11,
02482     12,    13,    -1,     6,     7,     8,     9,    10,    11,    12,
02483     13,    -1,     6,     7,     8,     9,    10,    11,    12,    13,
02484     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02485     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02486     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02487     -1,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
02488     -1,    -1,    -1,    -1,    -1,    -1,    78,    79,    80,    81,
02489     82,    83,    84,    -1,    86,    78,    79,    80,    81,    82,
02490     83,    84,    -1,    86,    78,    79,    80,    81,    82,    83,
02491     84,    -1,    86,    78,    79,    80,    81,    82,    83,    84,
02492     75,    86,    -1,    78,    79,    80,    81,    82,    83,    84,
02493     75,    -1,    -1,    78,    79,    80,    81,    82,    83,    84,
02494     75,    -1,    -1,    78,    79,    80,    81,    82,    83,    84,
02495      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02496      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02497      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02498      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02499      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02500      6,     7,     8,     9,    10,    11,    12,    13,    -1,    -1,
02501      6,     7,     8,     9,    10,    11,    12,    13,    -1,    75,
02502     -1,    -1,    78,    79,    80,    81,    82,    83,    84,    75,
02503     -1,    -1,    78,    79,    80,    81,    82,    83,    84,    75,
02504     -1,    -1,    78,    79,    80,    81,    82,    83,    84,    75,
02505     -1,    -1,    78,    79,    80,    81,    82,    83,    84,    75,
02506     -1,    -1,    78,    79,    80,    81,    82,    83,    84,    75,
02507     -1,    -1,    78,    79,    80,    81,    82,    83,    84,    75,
02508     -1,    -1,    78,    79,    80,    81,    82,    83,    84,     6,
02509      7,     8,     9,    10,    11,    12,    13,     6,     7,     8,
02510      9,    10,    11,    12,    13,    -1,    -1,     6,     7,     8,
02511      9,    10,    11,    12,    13,     6,     7,     8,     9,    10,
02512     11,    12,    13,    -1,    -1,    -1,    -1,     6,     7,     8,
02513      9,    10,    11,    12,    13,    -1,    45,     6,     7,     8,
02514      9,    10,    11,    12,    13,    -1,    -1,     6,     7,     8,
02515      9,    10,    11,    12,    13,    -1,    47,    -1,    75,    -1,
02516     -1,    78,    79,    80,    81,    82,    83,    84,    47,    78,
02517     79,    80,    81,    82,    83,    84,    75,    -1,    -1,    78,
02518     79,    80,    81,    82,    83,    84,    -1,    78,    79,    80,
02519     81,    82,    83,    84,    -1,    -1,    -1,    -1,    -1,    78,
02520     79,    80,    81,    82,    83,    84,    75,    -1,    -1,    78,
02521     79,    80,    81,    82,    83,    84,    75,    -1,    -1,    78,
02522     79,    80,    81,    82,    83,    84,     6,     7,     8,     9,
02523     10,    11,    12,    13,    -1,    -1,     6,     7,     8,     9,
02524     10,    11,    12,    13,    -1,    -1,     6,     7,     8,     9,
02525     10,    11,    12,    13,    -1,    -1,     6,     7,     8,     9,
02526     10,    11,    12,    13,    -1,    -1,     6,     7,     8,     9,
02527     10,    11,    12,    13,     6,     7,     8,     9,    10,    11,
```

```
    12,    13,    -1,    -1,     6,     7,     8,     9,    10,    11,
    12,    13,    -1,    -1,    -1,    75,    -1,    -1,    78,    79,
    80,    81,    82,    83,    84,    75,    -1,    -1,    78,    79,
    80,    81,    82,    83,    84,    75,    48,    -1,    78,    79,
    80,    81,    82,    83,    84,    75,    48,    -1,    78,    79,
    80,    81,    82,    83,    84,    75,    -1,    -1,    78,    79,
    80,    81,    82,    83,    84,    -1,    78,    79,    80,    81,
    82,    83,    84,    -1,    -1,    -1,    78,    79,    80,    81,
    82,    83,    84,     6,     7,     8,     9,    10,    11,    12,
    13,     6,     7,     8,     9,    10,    11,    12,    13,     6,
     7,     8,     9,    10,    11,    12,    13,     6,     7,     8,
     9,    10,    11,    12,    13,     6,     7,     8,     9,    10,
    11,    12,    13,     6,     7,     8,     9,    10,    11,    12,
    13,    -1,    -1,     6,     7,     8,     9,    10,    11,    12,
    13,     6,     7,     8,     9,    10,    11,    12,    13,    -1,
    -1,    -1,    75,    -1,    -1,    78,    79,    80,    81,    82,
    83,    84,    77,    78,    79,    80,    81,    82,    83,    84,
    77,    78,    79,    80,    81,    82,    83,    84,    77,    78,
    79,    80,    81,    82,    83,    84,    77,    78,    79,    80,
    81,    82,    83,    84,    77,    78,    79,    80,    81,    82,
    83,    84,    75,    -1,    -1,    78,    79,    80,    81,    82,
    83,    84,    77,    78,    79,    80,    81,    82,    83,    84,
     6,     7,     8,     9,    10,    11,    12,    13,     6,     7,
     8,     9,    10,    11,    12,    13,    -1,    -1,    -1,    -1,
    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,    -1,
    -1,    77,    78,    79,    80,    81,    82,    83,    84,    -1,
    78,    79,    80,    81,    82,    83,    84
};

const signed char
genesyspp_parser::yystos_[] =
{
     0,     1,     3,     4,     5,    14,    15,    16,    17,    18,
    19,    20,    21,    22,    23,    24,    28,    29,    30,    31,
    32,    33,    34,    35,    36,    37,    38,    39,    40,    44,
    46,    49,    50,    51,    54,    55,    56,    57,    58,    59,
    62,    63,    64,    65,    66,    69,    70,    71,    74,    79,
    89,    91,    92,    93,    94,    95,    96,    97,    98,    99,
   100,   101,   102,   103,   104,   105,   107,   108,   109,   110,
   111,    88,    92,    74,    74,    74,    74,    74,    74,    74,
    74,    74,    74,    74,    74,    74,    74,    74,    74,    74,
    74,    74,    74,    92,    49,    70,   107,   108,    76,    85,
    74,    74,    74,    74,    74,    74,    74,    74,    74,    74,
    74,    74,    74,    76,    85,    76,    92,    92,    74,     0,
     6,     7,     8,     9,    10,    11,    12,    13,    78,    79,
    80,    81,    82,    83,    84,    92,    92,    92,    92,    92,
    92,    92,    92,    92,    92,    92,    92,    92,    92,    92,
    92,    92,    92,    92,    92,   106,    79,    92,    76,    76,
    85,    85,    92,    92,    50,    53,    53,    53,    53,    53,
    68,    61,    61,    61,    61,    61,    68,    92,    92,    92,
    75,    92,    92,    92,    92,    92,    92,    92,    92,    92,
    92,    92,    92,    92,    92,    92,    92,    75,    75,    75,
    86,    75,    75,    75,    75,    75,    75,    75,    86,    86,
    86,    86,    86,    86,    86,    86,    86,    75,    86,    92,
    45,    92,    92,    92,    92,    77,    86,    75,    75,    75,
    75,    75,    75,    75,    75,    75,    75,    86,    86,    75,
    77,    86,    77,    86,    75,    92,    92,    92,    92,    92,
    92,    92,    92,    92,    92,    92,    92,    77,    86,    77,
    86,    47,    47,    85,    92,    49,     3,    85,    92,    92,
    75,    75,    75,    75,    75,    75,    75,    86,    86,    86,
    92,    92,    92,    92,    92,    77,    86,    75,    86,    92,
    77,    86,    77,    86,    92,    92,    92,    77,    86,    77,
    86,    48,    48,    85,    92,    49,    85,    92,    92,    75,
    86,    92,    92,    49,    70,   110,   110,    92,    77,    75,
    92,    77,    77,    92,    77,    77,    76,    76,    85,    85,
    75,    92,    92,    92,    92,    77,    86,    77,    86,    92,
    92,    77,    86,    77,    86,    92,    92,    77,    77
};

const signed char
genesyspp_parser::yyrl_[] =
{
     0,    90,    91,    91,    92,    92,    92,    92,    92,    92,
    92,    92,    92,    92,    92,    93,    93,    94,    94,    94,
    94,    94,    94,    95,    95,    95,    95,    95,    96,    96,
    96,    96,    96,    96,    97,    97,    98,    98,    99,    99,
   100,   100,   100,   100,   100,   100,   101,   101,   101,   101,
   102,   102,   103,   103,   103,   103,   103,   103,   103,   103,
   104,   104,   104,   104,   104,   104,   104,   104,   104,   104,
   104,   105,   106,   106,   107,   107,   107,   107,   108,   108,
   108,   108,   109,   109,   109,   109,   110,   110,   110,   110,
   110,   110,   110,   110,   111,   111,   111,   111,   111,   111,
   111,   111,   111,   111,   111,   111,   111
```

```
  };

  const signed char
  genesyspp_parser::yyr2_[] =
  {
       0,     2,     1,     2,     1,     1,     1,     1,     1,     1,
       1,     3,     1,     1,     1,     1,     1,     3,     3,     3,
       3,     3,     2,     3,     3,     3,     3,     2,     3,     3,
       3,     3,     3,     3,     1,     1,     5,     3,     8,     8,
       1,     1,     1,     1,     1,     1,     1,     1,     1,     4,
       4,     4,     4,     4,     4,     4,     4,     4,     4,     6,
       1,     4,     6,     6,     6,     6,     6,     6,     8,    10,
       4,     4,     5,     3,     1,     4,     6,     8,     1,     4,
       6,     8,     1,     4,     6,     8,     3,     6,     8,    10,
       3,     6,     8,    10,     1,     4,     4,     4,     6,     8,
       4,     4,     4,     4,     4,     4,     4
  };


#if YYDEBUG || 1
  // YYTNAME[SYMBOL-NUM] -- String name of the symbol SYMBOL-NUM.
  // First, the terminals, then, starting at \a YYNTOKENS, nonterminals.
  const char*
  const genesyspp_parser::yytname_[] =
  {
  "\"end of file\"", "error", "\"invalid token\"", "NUMD", "NUMH",
  "CTEZERO", "oLE", "oGE", "oEQ", "oNE", "oAND", "oOR", "oNAND", "oXOR",
  "oNOT", "fSIN", "fCOS", "fROUND", "fMOD", "fTRUNC", "fFRAC", "fEXP",
  "fSQRT", "fLOG", "fLN", "fVAL", "fEVAL", "fLENG", "fRND1", "fEXPO",
  "fNORM", "fUNIF", "fWEIB", "fLOGN", "fGAMM", "fERLA", "fTRIA", "fBETA",
  "fDISC", "fTNOW", "fTFIN", "fMAXREP", "fNUMREP", "fIDENT", "cIF",
  "cELSE", "cFOR", "cTO", "cDO", "ATRIB", "CSTAT", "fTAVG", "ILLEGAL",
  "RESOURCE", "fNR", "fMR", "fIRF", "fRESSEIZES", "fSTATE", "fSETSUM",
  "fRESUTIL", "QUEUE", "fNQ", "fFIRSTINQ", "fLASTINQ", "fSAQUE", "fAQUE",
  "fENTATRANK", "SET", "fNUMSET", "VARI", "FORM", "fNUMGR", "fATRGR",
  "\"(\"", "\")\"", "\"[\"", "\"]\"", "\"+\"", "\"-\"", "\"*\"", "\"^\"",
  "\"/\"", "\"<\"", "\">\"", "\"=\"", "\",\"", "NEG", "'\\n'", "\"USER\"",
  "$accept", "input", "expression", "numero", "aritmetica", "logica",
  "relacional", "comando", "comandoIF", "comandoFOR", "function",
  "kernelFunction", "trigonFunction", "mathFunction", "probFunction",
  "userFunction", "listaparm", "atributo", "variavel", "formula",
  "atribuicao", "pluginFunction", YY_NULLPTR
  };
#endif


#if YYDEBUG
  const short
  genesyspp_parser::yyrline_[] =
  {
       0,   225,   225,   226,   230,   231,   232,   233,   234,   235,
     236,   237,   238,   243,   247,   254,   255,   259,   260,   261,
     262,   263,   264,   268,   269,   270,   271,   272,   276,   277,
     278,   279,   280,   281,   285,   286,   290,   291,   296,   297,
     301,   302,   303,   304,   305,   306,   310,   311,   312,   313,
     320,   321,   325,   326,   327,   328,   329,   330,   331,   332,
     336,   337,   338,   339,   340,   341,   342,   343,   344,   345,
     346,   352,   357,   358,   384,   393,   402,   411,   425,   426,
     429,   432,   440,   442,   445,   448,   456,   459,   463,   467,
     473,   476,   480,   484,   495,   498,   500,   501,   508,   513,
     521,   522,   523,   524,   525,   526,   542
  };

  void
  genesyspp_parser::yy_stack_print_ () const
  {
    *yycdebug_ « "Stack now";
    for (stack_type::const_iterator
           i = yystack_.begin (),
           i_end = yystack_.end ();
         i != i_end; ++i)
      *yycdebug_ « ' ' « int (i->state);
    *yycdebug_ « '\n';
  }

  void
  genesyspp_parser::yy_reduce_print_ (int yyrule) const
  {
    int yylno = yyrline_[yyrule];
    int yynrhs = yyr2_[yyrule];
    // Print the symbols being reduced, and their result.
    *yycdebug_ « "Reducing stack by rule " « yyrule - 1
               « " (line " « yylno « "):\n";
    // The symbols being reduced.
    for (int yyi = 0; yyi < yynrhs; yyi++)
      YY_SYMBOL_PRINT ("   $" « yyi + 1 « " =",
                       yystack_[(yynrhs) - (yyi + 1)]);
```

```
  }
#endif // YYDEBUG


} // yy
#line 2708 "../GenesysParser.cpp"

#line 549 "bisonparser.yy"

void
yy::genesyspp_parser::error (const location_type& l,
                             const std::string& m)
{
  driver.error (l, m);
}
```

8.239 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/↩ GenesysParser.h File Reference

```
#include <string>
#include <cmath>
#include "obj_t.h"
#include "../../util/Util.h"
#include "../Attribute.h"
#include "../../../plugins/elements/Variable.h"
#include "../../../plugins/elements/Queue.h"
#include "../../../plugins/elements/Formula.h"
#include "../../../plugins/elements/Resource.h"
#include "../../../plugins/elements/Set.h"
#include <cassert>
#include <cstdlib>
#include <iostream>
#include <stdexcept>
#include <vector>
#include "location.hh"
#include <typeinfo>
```

Include dependency graph for GenesysParser.h:

This graph shows which files directly or indirectly include this file:

Classes

- class yy::genesyspp_parser

  A Bison parser.
- class yy::genesyspp_parser::semantic_type

- struct yy::genesyspp_parser::syntax_error

  Syntax errors thrown from user actions.
- struct yy::genesyspp_parser::token

  Token kinds.
- struct yy::genesyspp_parser::symbol_kind

  Symbol kinds.
- struct yy::genesyspp_parser::basic_symbol< Base >
- struct yy::genesyspp_parser::by_kind

  Type access provider for token (enum) based symbols.
- struct yy::genesyspp_parser::symbol_type

  "External" symbols: returned by the scanner.
- class yy::genesyspp_parser::context
- class yy::genesyspp_parser::stack< T, S >::slice

  Present a slice of the top of a stack.

Namespaces

- yy

Macros

- #define YY_CPLUSPLUS 199711L
- #define YY_MOVE
- #define YY_MOVE_OR_COPY copy
- #define YY_MOVE_REF(Type) Type&
- #define YY_RVREF(Type) const Type&
- #define YY_COPY(Type) const Type&
- #define YY_NOEXCEPT
- #define YY_NOTHROW throw ()
- #define YY_CONSTEXPR
- #define YY_ASSERT assert
- #define YY_ATTRIBUTE_PURE
- #define YY_ATTRIBUTE_UNUSED
- #define YY_USE(E) ((void) (E))
- #define YY_INITIAL_VALUE(Value) Value
- #define YY_IGNORE_MAYBE_UNINITIALIZED_BEGIN
- #define YY_IGNORE_MAYBE_UNINITIALIZED_END
- #define YY_IGNORE_USELESS_CAST_BEGIN
- #define YY_IGNORE_USELESS_CAST_END
- #define YY_CAST(Type, Val) ((Type) (Val))
- #define YY_REINTERPRET_CAST(Type, Val) ((Type) (Val))
- #define YYDEBUG 1

8.239.1 Detailed Description

Define the yy::parser class.

Definition in file GenesysParser.h.

8.239.2 Macro Definition Documentation

8.239.2.1 YY_ASSERT `#define YY_ASSERT assert`

Definition at line 131 of file GenesysParser.h.

8.239.2.2 YY_ATTRIBUTE_PURE `#define YY_ATTRIBUTE_PURE`

Definition at line 139 of file GenesysParser.h.

8.239.2.3 YY_ATTRIBUTE_UNUSED `#define YY_ATTRIBUTE_UNUSED`

Definition at line 147 of file GenesysParser.h.

8.239.2.4 YY_CAST

```
#define YY_CAST(
    Type,
    Val ) ((Type) (Val))
```

Definition at line 194 of file GenesysParser.h.

8.239.2.5 YY_CONSTEXPR `#define YY_CONSTEXPR`

Definition at line 125 of file GenesysParser.h.

8.239.2.6 YY_COPY

```
#define YY_COPY(
    Type ) const Type&
```

Definition at line 109 of file GenesysParser.h.

8.239.2.7 YY_CPLUSPLUS `#define YY_CPLUSPLUS 199711L`

Definition at line 94 of file GenesysParser.h.

8.239.2.8 YY_IGNORE_MAYBE_UNINITIALIZED_BEGIN `#define YY_IGNORE_MAYBE_UNINITIALIZED_BEGIN`

Definition at line 170 of file GenesysParser.h.

8.239.2.9 YY_IGNORE_MAYBE_UNINITIALIZED_END `#define YY_IGNORE_MAYBE_UNINITIALIZED_END`

Definition at line 171 of file GenesysParser.h.

8.239.2.10 YY_IGNORE_USELESS_CAST_BEGIN `#define YY_IGNORE_USELESS_CAST_BEGIN`

Definition at line 185 of file GenesysParser.h.

8.239.2.11 YY_IGNORE_USELESS_CAST_END `#define YY_IGNORE_USELESS_CAST_END`

Definition at line 186 of file GenesysParser.h.

8.239.2.12 YY_INITIAL_VALUE `#define YY_INITIAL_VALUE( Value ) Value`

Definition at line 167 of file GenesysParser.h.

8.239.2.13 YY_MOVE `#define YY_MOVE`

Definition at line 105 of file GenesysParser.h.

8.239.2.14 YY_MOVE_OR_COPY `#define YY_MOVE_OR_COPY copy`

Definition at line 106 of file GenesysParser.h.

8.239.2.15 YY_MOVE_REF `#define YY_MOVE_REF( Type ) Type&`

Definition at line 107 of file GenesysParser.h.

8.239.2.16 YY_NOEXCEPT `#define YY_NOEXCEPT`

Definition at line 117 of file GenesysParser.h.

8.239.2.17 YY_NOTHROW `#define YY_NOTHROW throw ()`

Definition at line 118 of file GenesysParser.h.

8.239.2.18 YY_REINTERPRET_CAST

```
#define YY_REINTERPRET_CAST(
    Type,
    Val ) ((Type) (Val))
```

Definition at line 195 of file GenesysParser.h.

8.239.2.19 YY_RVREF

```
#define YY_RVREF(
    Type ) const Type&
```

Definition at line 108 of file GenesysParser.h.

8.239.2.20 YY_USE

```
#define YY_USE(
    E ) ((void) (E))
```

Definition at line 153 of file GenesysParser.h.

8.239.2.21 YYDEBUG `#define YYDEBUG 1`

Definition at line 212 of file GenesysParser.h.

8.240 GenesysParser.h

```
00001 // A Bison parser, made by GNU Bison 3.7.6.
00002
00003 // Skeleton interface for Bison LALR(1) parsers in C++
00004
00005 // Copyright (C) 2002-2015, 2018-2021 Free Software Foundation, Inc.
00006
00007 // This program is free software: you can redistribute it and/or modify
00008 // it under the terms of the GNU General Public License as published by
00009 // the Free Software Foundation, either version 3 of the License, or
00010 // (at your option) any later version.
00011
00012 // This program is distributed in the hope that it will be useful,
00013 // but WITHOUT ANY WARRANTY; without even the implied warranty of
00014 // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
00015 // GNU General Public License for more details.
00016
00017 // You should have received a copy of the GNU General Public License
00018 // along with this program.  If not, see <https://www.gnu.org/licenses/>.
00019
00020 // As a special exception, you may create a larger work that contains
00021 // part or all of the Bison parser skeleton and distribute that work
00022 // under terms of your choice, so long as that work isn't itself a
00023 // parser generator using the skeleton or a modified version thereof
00024 // as a parser skeleton.  Alternatively, if you modify or redistribute
00025 // the parser skeleton itself, you may (at your option) remove this
00026 // special exception, which will cause the skeleton and the resulting
00027 // Bison output files to be licensed under the GNU General Public
00028 // License without this special exception.
00029
00030 // This special exception was added by the Free Software Foundation in
00031 // version 2.2 of Bison.
00032
00033
00039 // C++ LALR(1) parser skeleton written by Akim Demaille.
00040
00041 // DO NOT RELY ON FEATURES THAT ARE NOT DOCUMENTED in the manual,
00042 // especially those whose name start with YY_ or yy_.  They are
00043 // private implementation details that can be changed or removed.
00044
00045 #ifndef YY_YY_GENESYSPARSER_H_INCLUDED
00046 # define YY_YY_GENESYSPARSER_H_INCLUDED
00047 // "%code requires" blocks.
00048 #line 9 "bisonparser.yy"
00049
00050     #include <string>
00051     #include <cmath>
00052     #include "obj_t.h"
00053     #include "../../util/Util.h"
00054     #include "../Attribute.h"
00055
00056     /****begin_Includes_plugins****/
00057
00059         #include "../../../plugins/elements/Variable.h"
00063         #include "../../../plugins/elements/Queue.h"
00067         #include "../../../plugins/elements/Formula.h"
00071         #include "../../../plugins/elements/Resource.h"
00075         #include "../../../plugins/elements/Set.h"
00078     /****end_Includes_plugins****/
00079
00080     class genesyspp_driver;
00081
00082 #line 83 "../GenesysParser.h"
00083
00084 # include <cassert>
00085 # include <cstdlib> // std::abort
00086 # include <iostream>
00087 # include <stdexcept>
00088 # include <string>
00089 # include <vector>
00090
00091 #if defined __cplusplus
00092 # define YY_CPLUSPLUS __cplusplus
00093 #else
00094 # define YY_CPLUSPLUS 199711L
00095 #endif
00096
00097 // Support move semantics when possible.
00098 #if 201103L <= YY_CPLUSPLUS
00099 # define YY_MOVE           std::move
00100 # define YY_MOVE_OR_COPY   move
00101 # define YY_MOVE_REF(Type) Type&&
00102 # define YY_RVREF(Type)    Type&&
00103 # define YY_COPY(Type)     Type
00104 #else
00105 # define YY_MOVE
```

```
# define YY_MOVE_OR_COPY   copy
# define YY_MOVE_REF(Type) Type&
# define YY_RVREF(Type)    const Type&
# define YY_COPY(Type)     const Type&
#endif

// Support noexcept when possible.
#if 201103L <= YY_CPLUSPLUS
# define YY_NOEXCEPT noexcept
# define YY_NOTHROW
#else
# define YY_NOEXCEPT
# define YY_NOTHROW throw ()
#endif

// Support constexpr when possible.
#if 201703 <= YY_CPLUSPLUS
# define YY_CONSTEXPR constexpr
#else
# define YY_CONSTEXPR
#endif
# include "location.hh"
#include <typeinfo>
#ifndef YY_ASSERT
# include <cassert>
# define YY_ASSERT assert
#endif


#ifndef YY_ATTRIBUTE_PURE
# if defined __GNUC__ && 2 < __GNUC__ + (96 <= __GNUC_MINOR__)
#  define YY_ATTRIBUTE_PURE __attribute__ ((__pure__))
# else
#  define YY_ATTRIBUTE_PURE
# endif
#endif

#ifndef YY_ATTRIBUTE_UNUSED
# if defined __GNUC__ && 2 < __GNUC__ + (7 <= __GNUC_MINOR__)
#  define YY_ATTRIBUTE_UNUSED __attribute__ ((__unused__))
# else
#  define YY_ATTRIBUTE_UNUSED
# endif
#endif

/* Suppress unused-variable warnings by "using" E.  */
#if ! defined lint || defined __GNUC__
# define YY_USE(E) ((void) (E))
#else
# define YY_USE(E) /* empty */
#endif

#if defined __GNUC__ && ! defined __ICC && 407 <= __GNUC__ * 100 + __GNUC_MINOR__
/* Suppress an incorrect diagnostic about yylval being uninitialized.  */
# define YY_IGNORE_MAYBE_UNINITIALIZED_BEGIN                            \
    _Pragma ("GCC diagnostic push")                                     \
    _Pragma ("GCC diagnostic ignored \"-Wuninitialized\"")              \
    _Pragma ("GCC diagnostic ignored \"-Wmaybe-uninitialized\"")
# define YY_IGNORE_MAYBE_UNINITIALIZED_END      \
    _Pragma ("GCC diagnostic pop")
#else
# define YY_INITIAL_VALUE(Value) Value
#endif
#ifndef YY_IGNORE_MAYBE_UNINITIALIZED_BEGIN
# define YY_IGNORE_MAYBE_UNINITIALIZED_BEGIN
# define YY_IGNORE_MAYBE_UNINITIALIZED_END
#endif
#ifndef YY_INITIAL_VALUE
# define YY_INITIAL_VALUE(Value) /* Nothing. */
#endif

#if defined __cplusplus && defined __GNUC__ && ! defined __ICC && 6 <= __GNUC__
# define YY_IGNORE_USELESS_CAST_BEGIN                          \
    _Pragma ("GCC diagnostic push")                            \
    _Pragma ("GCC diagnostic ignored \"-Wuseless-cast\"")
# define YY_IGNORE_USELESS_CAST_END            \
    _Pragma ("GCC diagnostic pop")
#endif
#ifndef YY_IGNORE_USELESS_CAST_BEGIN
# define YY_IGNORE_USELESS_CAST_BEGIN
# define YY_IGNORE_USELESS_CAST_END
#endif

# ifndef YY_CAST
#  ifdef __cplusplus
#   define YY_CAST(Type, Val) static_cast<Type> (Val)
#   define YY_REINTERPRET_CAST(Type, Val) reinterpret_cast<Type> (Val)
```

```
#  else
#   define YY_CAST(Type, Val) ((Type) (Val))
#   define YY_REINTERPRET_CAST(Type, Val) ((Type) (Val))
#  endif
# endif
# ifndef YY_NULLPTR
#  if defined __cplusplus
#   if 201103L <= __cplusplus
#    define YY_NULLPTR nullptr
#   else
#    define YY_NULLPTR 0
#   endif
#  else
#   define YY_NULLPTR ((void*)0)
#  endif
# endif

/* Debug traces.  */
#ifndef YYDEBUG
# define YYDEBUG 1
#endif

namespace yy {
#line 217 "../GenesysParser.h"



  class genesyspp_parser
  {
  public:
#ifndef YYSTYPE
  class semantic_type
  {
  public:
    typedef semantic_type self_type;

    semantic_type () YY_NOEXCEPT
      : yybuffer_ ()
      , yytypeid_ (YY_NULLPTR)
    {}

    template <typename T>
    semantic_type (YY_RVREF (T) t)
      : yytypeid_ (&typeid (T))
    {
      YY_ASSERT (sizeof (T) <= size);
      new (yyas_<T> ()) T (YY_MOVE (t));
    }

#if 201103L <= YY_CPLUSPLUS
    semantic_type (const self_type&) = delete;
    self_type& operator= (const self_type&) = delete;
#endif

    ~semantic_type () YY_NOEXCEPT
    {
      YY_ASSERT (!yytypeid_);
    }

# if 201103L <= YY_CPLUSPLUS
    template <typename T, typename... U>
    T&
    emplace (U&&... u)
    {
      YY_ASSERT (!yytypeid_);
      YY_ASSERT (sizeof (T) <= size);
      yytypeid_ = & typeid (T);
      return *new (yyas_<T> ()) T (std::forward <U>(u)...);
    }
# else
    template <typename T>
    T&
    emplace ()
    {
      YY_ASSERT (!yytypeid_);
      YY_ASSERT (sizeof (T) <= size);
      yytypeid_ = & typeid (T);
      return *new (yyas_<T> ()) T ();
    }

    template <typename T>
    T&
    emplace (const T& t)
    {
      YY_ASSERT (!yytypeid_);
      YY_ASSERT (sizeof (T) <= size);
```

```
      yytypeid_ = & typeid (T);
      return *new (yyas_<T> ()) T (t);
    }
# endif

    template <typename T>
    T&
    build ()
    {
      return emplace<T> ();
    }

    template <typename T>
    T&
    build (const T& t)
    {
      return emplace<T> (t);
    }

    template <typename T>
    T&
    as () YY_NOEXCEPT
    {
      YY_ASSERT (yytypeid_);
      YY_ASSERT (*yytypeid_ == typeid (T));
      YY_ASSERT (sizeof (T) <= size);
      return *yyas_<T> ();
    }

    template <typename T>
    const T&
    as () const YY_NOEXCEPT
    {
      YY_ASSERT (yytypeid_);
      YY_ASSERT (*yytypeid_ == typeid (T));
      YY_ASSERT (sizeof (T) <= size);
      return *yyas_<T> ();
    }

    template <typename T>
    void
    swap (self_type& that) YY_NOEXCEPT
    {
      YY_ASSERT (yytypeid_);
      YY_ASSERT (*yytypeid_ == *that.yytypeid_);
      std::swap (as<T> (), that.as<T> ());
    }

    template <typename T>
    void
    move (self_type& that)
    {
# if 201103L <= YY_CPLUSPLUS
      emplace<T> (std::move (that.as<T> ()));
# else
      emplace<T> ();
      swap<T> (that);
# endif
      that.destroy<T> ();
    }

# if 201103L <= YY_CPLUSPLUS
    template <typename T>
    void
    move (self_type&& that)
    {
      emplace<T> (std::move (that.as<T> ()));
      that.destroy<T> ();
    }
#endif

    template <typename T>
    void
    copy (const self_type& that)
    {
      emplace<T> (that.as<T> ());
    }

    template <typename T>
    void
    destroy ()
    {
      as<T> ().~T ();
      yytypeid_ = YY_NULLPTR;
    }

  private:
```

```
#if YY_CPLUSPLUS < 201103L
    semantic_type (const self_type&);
    self_type& operator= (const self_type&);
#endif

    template <typename T>
    T*
    yyas_ () YY_NOEXCEPT
    {
      void *yyp = yybuffer_.yyraw;
      return static_cast<T*> (yyp);
     }

    template <typename T>
    const T*
    yyas_ () const YY_NOEXCEPT
    {
      const void *yyp = yybuffer_.yyraw;
      return static_cast<const T*> (yyp);
     }

    union union_type
    {
      // NUMD
      // NUMH
      // CTEZERO
      // oLE
      // oGE
      // oEQ
      // oNE
      // oAND
      // oOR
      // oNAND
      // oXOR
      // oNOT
      // fSIN
      // fCOS
      // fROUND
      // fMOD
      // fTRUNC
      // fFRAC
      // fEXP
      // fSQRT
      // fLOG
      // fLN
      // fVAL
      // fEVAL
      // fLENG
      // fRND1
      // fEXPO
      // fNORM
      // fUNIF
      // fWEIB
      // fLOGN
      // fGAMM
      // fERLA
      // fTRIA
      // fBETA
      // fDISC
      // fTNOW
      // fTFIN
      // fMAXREP
      // fNUMREP
      // fIDENT
      // cIF
      // cELSE
      // cFOR
      // cTO
      // cDO
      // ATRIB
      // CSTAT
      // fTAVG
      // ILLEGAL
      // RESOURCE
      // fNR
      // fMR
      // fIRF
      // fRESSEIZES
      // fSTATE
      // fSETSUM
      // fRESUTIL
      // QUEUE
      // fNQ
      // fFIRSTINQ
      // fLASTINQ
      // fSAQUE
      // fAQUE
```

```
      // fENTATRANK
      // SET
      // fNUMSET
      // VARI
      // FORM
      // fNUMGR
      // fATRGR
      // input
      // expression
      // numero
      // aritmetica
      // logica
      // relacional
      // comando
      // comandoIF
      // comandoFOR
      // function
      // kernelFunction
      // trigonFunction
      // mathFunction
      // probFunction
      // userFunction
      // listaparm
      // atributo
      // variavel
      // formula
      // atribuicao
      // pluginFunction
      char dummy1[sizeof (obj_t)];
    };

    enum { size = sizeof (union_type) };

    union
    {
      long double yyalign_me;
      char yyraw[size];
    } yybuffer_;

    const std::type_info *yytypeid_;
  };

#else
    typedef YYSTYPE semantic_type;
#endif
    typedef location location_type;

    struct syntax_error : std::runtime_error
    {
      syntax_error (const location_type& l, const std::string& m)
        : std::runtime_error (m)
        , location (l)
      {}

      syntax_error (const syntax_error& s)
        : std::runtime_error (s.what ())
        , location (s.location)
      {}

      ~syntax_error () YY_NOEXCEPT YY_NOTHROW;

      location_type location;
    };

    struct token
    {
      enum token_kind_type
      {
        YYEMPTY = -2,
    END = 0,                        // "end of file"
    YYerror = 256,                  // error
    YYUNDEF = 257,                  // "invalid token"
    NUMD = 258,                     // NUMD
    NUMH = 259,                     // NUMH
    CTEZERO = 260,                  // CTEZERO
    oLE = 261,                      // oLE
    oGE = 262,                      // oGE
    oEQ = 263,                      // oEQ
    oNE = 264,                      // oNE
    oAND = 265,                     // oAND
    oOR = 266,                      // oOR
    oNAND = 267,                    // oNAND
    oXOR = 268,                     // oXOR
    oNOT = 269,                     // oNOT
    fSIN = 270,                     // fSIN
    fCOS = 271,                     // fCOS
    fROUND = 272,                   // fROUND
```

```
    fMOD = 273,                     // fMOD
    fTRUNC = 274,                   // fTRUNC
    fFRAC = 275,                    // fFRAC
    fEXP = 276,                     // fEXP
    fSQRT = 277,                    // fSQRT
    fLOG = 278,                     // fLOG
    fLN = 279,                      // fLN
    fVAL = 280,                     // fVAL
    fEVAL = 281,                    // fEVAL
    fLENG = 282,                    // fLENG
    fRND1 = 283,                    // fRND1
    fEXPO = 284,                    // fEXPO
    fNORM = 285,                    // fNORM
    fUNIF = 286,                    // fUNIF
    fWEIB = 287,                    // fWEIB
    fLOGN = 288,                    // fLOGN
    fGAMM = 289,                    // fGAMM
    fERLA = 290,                    // fERLA
    fTRIA = 291,                    // fTRIA
    fBETA = 292,                    // fBETA
    fDISC = 293,                    // fDISC
    fTNOW = 294,                    // fTNOW
    fTFIN = 295,                    // fTFIN
    fMAXREP = 296,                  // fMAXREP
    fNUMREP = 297,                  // fNUMREP
    fIDENT = 298,                   // fIDENT
    cIF = 299,                      // cIF
    cELSE = 300,                    // cELSE
    cFOR = 301,                     // cFOR
    cTO = 302,                      // cTO
    cDO = 303,                      // cDO
    ATRIB = 304,                    // ATRIB
    CSTAT = 305,                    // CSTAT
    fTAVG = 306,                    // fTAVG
    ILLEGAL = 307,                  // ILLEGAL
    RESOURCE = 308,                 // RESOURCE
    fNR = 309,                      // fNR
    fMR = 310,                      // fMR
    fIRF = 311,                     // fIRF
    fRESSEIZES = 312,               // fRESSEIZES
    fSTATE = 313,                   // fSTATE
    fSETSUM = 314,                  // fSETSUM
    fRESUTIL = 315,                 // fRESUTIL
    QUEUE = 316,                    // QUEUE
    fNQ = 317,                      // fNQ
    fFIRSTINQ = 318,                // fFIRSTINQ
    fLASTINQ = 319,                 // fLASTINQ
    fSAQUE = 320,                   // fSAQUE
    fAQUE = 321,                    // fAQUE
    fENTATRANK = 322,               // fENTATRANK
    SET = 323,                      // SET
    fNUMSET = 324,                  // fNUMSET
    VARI = 325,                     // VARI
    FORM = 326,                     // FORM
    fNUMGR = 327,                   // fNUMGR
    fATRGR = 328,                   // fATRGR
    LPAREN = 329,                   // "("
    RPAREN = 330,                   // ")"
    LBRACKET = 331,                 // "["
    RBRACKET = 332,                 // "]"
    PLUS = 333,                     // "+"
    MINUS = 334,                    // "-"
    STAR = 335,                     // "*"
    POWER = 336,                    // "^"
    SLASH = 337,                    // "/"
    LESS = 338,                     // "<"
    GREATER = 339,                  // ">"
    ASSIGN = 340,                   // "="
    COMMA = 341,                    // ","
    NEG = 342                       // NEG
      };
      typedef token_kind_type yytokentype;
    };

    typedef token::yytokentype token_kind_type;

    typedef token_kind_type token_type;

    struct symbol_kind
    {
      enum symbol_kind_type
      {
        YYNTOKENS = 90,
        S_YYEMPTY = -2,
        S_YYEOF = 0,                            // "end of file"
        S_YYerror = 1,                          // error
        S_YYUNDEF = 2,                          // "invalid token"
```

```
        S_NUMD = 3,                          // NUMD
        S_NUMH = 4,                          // NUMH
        S_CTEZERO = 5,                       // CTEZERO
        S_oLE = 6,                           // oLE
        S_oGE = 7,                           // oGE
        S_oEQ = 8,                           // oEQ
        S_oNE = 9,                           // oNE
        S_oAND = 10,                         // oAND
        S_oOR = 11,                          // oOR
        S_oNAND = 12,                        // oNAND
        S_oXOR = 13,                         // oXOR
        S_oNOT = 14,                         // oNOT
        S_fSIN = 15,                         // fSIN
        S_fCOS = 16,                         // fCOS
        S_fROUND = 17,                       // fROUND
        S_fMOD = 18,                         // fMOD
        S_fTRUNC = 19,                       // fTRUNC
        S_fFRAC = 20,                        // fFRAC
        S_fEXP = 21,                         // fEXP
        S_fSQRT = 22,                        // fSQRT
        S_fLOG = 23,                         // fLOG
        S_fLN = 24,                          // fLN
        S_fVAL = 25,                         // fVAL
        S_fEVAL = 26,                        // fEVAL
        S_fLENG = 27,                        // fLENG
        S_fRND1 = 28,                        // fRND1
        S_fEXPO = 29,                        // fEXPO
        S_fNORM = 30,                        // fNORM
        S_fUNIF = 31,                        // fUNIF
        S_fWEIB = 32,                        // fWEIB
        S_fLOGN = 33,                        // fLOGN
        S_fGAMM = 34,                        // fGAMM
        S_fERLA = 35,                        // fERLA
        S_fTRIA = 36,                        // fTRIA
        S_fBETA = 37,                        // fBETA
        S_fDISC = 38,                        // fDISC
        S_fTNOW = 39,                        // fTNOW
        S_fTFIN = 40,                        // fTFIN
        S_fMAXREP = 41,                      // fMAXREP
        S_fNUMREP = 42,                      // fNUMREP
        S_fIDENT = 43,                       // fIDENT
        S_cIF = 44,                          // cIF
        S_cELSE = 45,                        // cELSE
        S_cFOR = 46,                         // cFOR
        S_cTO = 47,                          // cTO
        S_cDO = 48,                          // cDO
        S_ATRIB = 49,                        // ATRIB
        S_CSTAT = 50,                        // CSTAT
        S_fTAVG = 51,                        // fTAVG
        S_ILLEGAL = 52,                      // ILLEGAL
        S_RESOURCE = 53,                     // RESOURCE
        S_fNR = 54,                          // fNR
        S_fMR = 55,                          // fMR
        S_fIRF = 56,                         // fIRF
        S_fRESSEIZES = 57,                   // fRESSEIZES
        S_fSTATE = 58,                       // fSTATE
        S_fSETSUM = 59,                      // fSETSUM
        S_fRESUTIL = 60,                     // fRESUTIL
        S_QUEUE = 61,                        // QUEUE
        S_fNQ = 62,                          // fNQ
        S_fFIRSTINQ = 63,                    // fFIRSTINQ
        S_fLASTINQ = 64,                     // fLASTINQ
        S_fSAQUE = 65,                       // fSAQUE
        S_fAQUE = 66,                        // fAQUE
        S_fENTATRANK = 67,                   // fENTATRANK
        S_SET = 68,                          // SET
        S_fNUMSET = 69,                      // fNUMSET
        S_VARI = 70,                         // VARI
        S_FORM = 71,                         // FORM
        S_fNUMGR = 72,                       // fNUMGR
        S_fATRGR = 73,                       // fATRGR
        S_LPAREN = 74,                       // "("
        S_RPAREN = 75,                       // ")"
        S_LBRACKET = 76,                     // "["
        S_RBRACKET = 77,                     // "]"
        S_PLUS = 78,                         // "+"
        S_MINUS = 79,                        // "-"
        S_STAR = 80,                         // "*"
        S_POWER = 81,                        // "^"
        S_SLASH = 82,                        // "/"
        S_LESS = 83,                         // "<"
        S_GREATER = 84,                      // ">"
        S_ASSIGN = 85,                       // "="
        S_COMMA = 86,                        // ","
        S_NEG = 87,                          // NEG
        S_88_n_ = 88,                        // '\n'
        S_89_USER_ = 89,                     // "USER"
```

```
        S_YYACCEPT = 90,                        // $accept
        S_input = 91,                           // input
        S_expression = 92,                      // expression
        S_numero = 93,                          // numero
        S_aritmetica = 94,                      // aritmetica
        S_logica = 95,                          // logica
        S_relacional = 96,                      // relacional
        S_comando = 97,                         // comando
        S_comandoIF = 98,                       // comandoIF
        S_comandoFOR = 99,                      // comandoFOR
        S_function = 100,                       // function
        S_kernelFunction = 101,                 // kernelFunction
        S_trigonFunction = 102,                 // trigonFunction
        S_mathFunction = 103,                   // mathFunction
        S_probFunction = 104,                   // probFunction
        S_userFunction = 105,                   // userFunction
        S_listaparm = 106,                      // listaparm
        S_atributo = 107,                       // atributo
        S_variavel = 108,                       // variavel
        S_formula = 109,                        // formula
        S_atribuicao = 110,                     // atribuicao
        S_pluginFunction = 111                  // pluginFunction
      };
    };

    typedef symbol_kind::symbol_kind_type symbol_kind_type;

    static const symbol_kind_type YYNTOKENS = symbol_kind::YYNTOKENS;

    template <typename Base>
    struct basic_symbol : Base
    {
      typedef Base super_type;

      basic_symbol ()
        : value ()
        , location ()
      {}

#if 201103L <= YY_CPLUSPLUS
      basic_symbol (basic_symbol&& that)
        : Base (std::move (that))
        , value ()
        , location (std::move (that.location))
      {
        switch (this->kind ())
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
```

```
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.move< obj_t > (std::move (that.value));
        break;

      default:
        break;
    }

      }
#endif

      basic_symbol (const basic_symbol& that);

#if 201103L <= YY_CPLUSPLUS
      basic_symbol (typename Base::kind_type t, location_type&& l)
        : Base (t)
        , location (std::move (l))
      {}
#else
      basic_symbol (typename Base::kind_type t, const location_type& l)
        : Base (t)
        , location (l)
      {}
#endif

#if 201103L <= YY_CPLUSPLUS
      basic_symbol (typename Base::kind_type t, obj_t&& v, location_type&& l)
        : Base (t)
        , value (std::move (v))
        , location (std::move (l))
      {}
#else
      basic_symbol (typename Base::kind_type t, const obj_t& v, const location_type& l)
        : Base (t)
        , value (v)
        , location (l)
```

```
      {}
#endif

      ~basic_symbol ()
      {
        clear ();
      }

      void clear () YY_NOEXCEPT
      {
        // User destructor.
        symbol_kind_type yykind = this->kind ();
        basic_symbol<Base>& yysym = *this;
        (void) yysym;
        switch (yykind)
        {
       default:
          break;
        }

        // Value type destructor.
switch (yykind)
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
```

```
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.template destroy< obj_t > ();
        break;

      default:
        break;
    }

        Base::clear ();
      }

      std::string name () const YY_NOEXCEPT
      {
        return genesyspp_parser::symbol_name (this->kind ());
      }

      symbol_kind_type type_get () const YY_NOEXCEPT;

      bool empty () const YY_NOEXCEPT;

      void move (basic_symbol& s);

      semantic_type value;

      location_type location;

    private:
#if YY_CPLUSPLUS < 201103L
      basic_symbol& operator= (const basic_symbol& that);
#endif
    };

    struct by_kind
    {
      by_kind ();

#if 201103L <= YY_CPLUSPLUS
      by_kind (by_kind&& that);
#endif

      by_kind (const by_kind& that);

      typedef token_kind_type kind_type;

      by_kind (kind_type t);

      void clear () YY_NOEXCEPT;

      void move (by_kind& that);

      symbol_kind_type kind () const YY_NOEXCEPT;

      symbol_kind_type type_get () const YY_NOEXCEPT;

      symbol_kind_type kind_;
    };

    typedef by_kind by_type;

    struct symbol_type : basic_symbol<by_kind>
```

```
    {
      typedef basic_symbol<by_kind> super_type;

      symbol_type () {}

#if 201103L <= YY_CPLUSPLUS
      symbol_type (int tok, location_type l)
        : super_type(token_type (tok), std::move (l))
#else
      symbol_type (int tok, const location_type& l)
        : super_type(token_type (tok), l)
#endif
      {
        YY_ASSERT (tok == token::END
                   || (token::YYerror <= tok && tok <= token::YYUNDEF)
                   || (token::LPAREN <= tok && tok <= token::NEG)
                   || tok == 10
                   || tok == 343);
      }
#if 201103L <= YY_CPLUSPLUS
      symbol_type (int tok, obj_t v, location_type l)
        : super_type(token_type (tok), std::move (v), std::move (l))
#else
      symbol_type (int tok, const obj_t& v, const location_type& l)
        : super_type(token_type (tok), v, l)
#endif
      {
        YY_ASSERT ((token::NUMD <= tok && tok <= token::fATRGR));
      }
    };

    genesyspp_parser (genesyspp_driver& driver_yyarg);
    virtual ~genesyspp_parser ();

#if 201103L <= YY_CPLUSPLUS
    genesyspp_parser (const genesyspp_parser&) = delete;
    genesyspp_parser& operator= (const genesyspp_parser&) = delete;
#endif

    int operator() ();

    virtual int parse ();

#if YYDEBUG
    std::ostream& debug_stream () const YY_ATTRIBUTE_PURE;
    void set_debug_stream (std::ostream &);

    typedef int debug_level_type;
    debug_level_type debug_level () const YY_ATTRIBUTE_PURE;
    void set_debug_level (debug_level_type l);
#endif

    virtual void error (const location_type& loc, const std::string& msg);

    void error (const syntax_error& err);

    static std::string symbol_name (symbol_kind_type yysymbol);

    // Implementation of make_symbol for each symbol type.
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_END (location_type l)
      {
        return symbol_type (token::END, std::move (l));
      }
#else
      static
      symbol_type
      make_END (const location_type& l)
      {
        return symbol_type (token::END, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_YYerror (location_type l)
      {
        return symbol_type (token::YYerror, std::move (l));
      }
#else
      static
      symbol_type
      make_YYerror (const location_type& l)
      {
        return symbol_type (token::YYerror, l);
```

```
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_YYUNDEF (location_type l)
      {
        return symbol_type (token::YYUNDEF, std::move (l));
      }
#else
      static
      symbol_type
      make_YYUNDEF (const location_type& l)
      {
        return symbol_type (token::YYUNDEF, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_NUMD (obj_t v, location_type l)
      {
        return symbol_type (token::NUMD, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_NUMD (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::NUMD, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_NUMH (obj_t v, location_type l)
      {
        return symbol_type (token::NUMH, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_NUMH (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::NUMH, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_CTEZERO (obj_t v, location_type l)
      {
        return symbol_type (token::CTEZERO, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_CTEZERO (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::CTEZERO, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oLE (obj_t v, location_type l)
      {
        return symbol_type (token::oLE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oLE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oLE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oGE (obj_t v, location_type l)
      {
        return symbol_type (token::oGE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
```

```
      make_oGE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oGE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oEQ (obj_t v, location_type l)
      {
        return symbol_type (token::oEQ, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oEQ (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oEQ, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oNE (obj_t v, location_type l)
      {
        return symbol_type (token::oNE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oNE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oNE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oAND (obj_t v, location_type l)
      {
        return symbol_type (token::oAND, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oAND (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oAND, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oOR (obj_t v, location_type l)
      {
        return symbol_type (token::oOR, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oOR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oOR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oNAND (obj_t v, location_type l)
      {
        return symbol_type (token::oNAND, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oNAND (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oNAND, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oXOR (obj_t v, location_type l)
      {
        return symbol_type (token::oXOR, std::move (v), std::move (l));
      }
```

```
#else
      static
      symbol_type
      make_oXOR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oXOR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_oNOT (obj_t v, location_type l)
      {
        return symbol_type (token::oNOT, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_oNOT (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::oNOT, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fSIN (obj_t v, location_type l)
      {
        return symbol_type (token::fSIN, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fSIN (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fSIN, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fCOS (obj_t v, location_type l)
      {
        return symbol_type (token::fCOS, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fCOS (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fCOS, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fROUND (obj_t v, location_type l)
      {
        return symbol_type (token::fROUND, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fROUND (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fROUND, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fMOD (obj_t v, location_type l)
      {
        return symbol_type (token::fMOD, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fMOD (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fMOD, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fTRUNC (obj_t v, location_type l)
```

```
      {
        return symbol_type (token::fTRUNC, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fTRUNC (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fTRUNC, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fFRAC (obj_t v, location_type l)
      {
        return symbol_type (token::fFRAC, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fFRAC (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fFRAC, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fEXP (obj_t v, location_type l)
      {
        return symbol_type (token::fEXP, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fEXP (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fEXP, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fSQRT (obj_t v, location_type l)
      {
        return symbol_type (token::fSQRT, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fSQRT (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fSQRT, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fLOG (obj_t v, location_type l)
      {
        return symbol_type (token::fLOG, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fLOG (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fLOG, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fLN (obj_t v, location_type l)
      {
        return symbol_type (token::fLN, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fLN (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fLN, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
```

```
      static
      symbol_type
      make_fVAL (obj_t v, location_type l)
      {
        return symbol_type (token::fVAL, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fVAL (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fVAL, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fEVAL (obj_t v, location_type l)
      {
        return symbol_type (token::fEVAL, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fEVAL (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fEVAL, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fLENG (obj_t v, location_type l)
      {
        return symbol_type (token::fLENG, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fLENG (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fLENG, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fRND1 (obj_t v, location_type l)
      {
        return symbol_type (token::fRND1, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fRND1 (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fRND1, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fEXPO (obj_t v, location_type l)
      {
        return symbol_type (token::fEXPO, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fEXPO (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fEXPO, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fNORM (obj_t v, location_type l)
      {
        return symbol_type (token::fNORM, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fNORM (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fNORM, v, l);
```

```
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fUNIF (obj_t v, location_type l)
      {
        return symbol_type (token::fUNIF, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fUNIF (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fUNIF, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fWEIB (obj_t v, location_type l)
      {
        return symbol_type (token::fWEIB, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fWEIB (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fWEIB, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fLOGN (obj_t v, location_type l)
      {
        return symbol_type (token::fLOGN, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fLOGN (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fLOGN, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fGAMM (obj_t v, location_type l)
      {
        return symbol_type (token::fGAMM, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fGAMM (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fGAMM, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fERLA (obj_t v, location_type l)
      {
        return symbol_type (token::fERLA, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fERLA (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fERLA, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fTRIA (obj_t v, location_type l)
      {
        return symbol_type (token::fTRIA, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
```

```
      make_fTRIA (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fTRIA, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fBETA (obj_t v, location_type l)
      {
        return symbol_type (token::fBETA, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fBETA (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fBETA, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fDISC (obj_t v, location_type l)
      {
        return symbol_type (token::fDISC, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fDISC (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fDISC, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fTNOW (obj_t v, location_type l)
      {
        return symbol_type (token::fTNOW, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fTNOW (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fTNOW, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fTFIN (obj_t v, location_type l)
      {
        return symbol_type (token::fTFIN, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fTFIN (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fTFIN, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fMAXREP (obj_t v, location_type l)
      {
        return symbol_type (token::fMAXREP, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fMAXREP (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fMAXREP, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fNUMREP (obj_t v, location_type l)
      {
        return symbol_type (token::fNUMREP, std::move (v), std::move (l));
      }
```

```
#else
      static
      symbol_type
      make_fNUMREP (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fNUMREP, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fIDENT (obj_t v, location_type l)
      {
        return symbol_type (token::fIDENT, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fIDENT (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fIDENT, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_cIF (obj_t v, location_type l)
      {
        return symbol_type (token::cIF, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_cIF (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::cIF, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_cELSE (obj_t v, location_type l)
      {
        return symbol_type (token::cELSE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_cELSE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::cELSE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_cFOR (obj_t v, location_type l)
      {
        return symbol_type (token::cFOR, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_cFOR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::cFOR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_cTO (obj_t v, location_type l)
      {
        return symbol_type (token::cTO, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_cTO (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::cTO, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_cDO (obj_t v, location_type l)
```

```
      {
        return symbol_type (token::cDO, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_cDO (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::cDO, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_ATRIB (obj_t v, location_type l)
      {
        return symbol_type (token::ATRIB, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_ATRIB (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::ATRIB, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_CSTAT (obj_t v, location_type l)
      {
        return symbol_type (token::CSTAT, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_CSTAT (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::CSTAT, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fTAVG (obj_t v, location_type l)
      {
        return symbol_type (token::fTAVG, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fTAVG (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fTAVG, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_ILLEGAL (obj_t v, location_type l)
      {
        return symbol_type (token::ILLEGAL, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_ILLEGAL (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::ILLEGAL, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_RESOURCE (obj_t v, location_type l)
      {
        return symbol_type (token::RESOURCE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_RESOURCE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::RESOURCE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
```

```
      static
      symbol_type
      make_fNR (obj_t v, location_type l)
      {
        return symbol_type (token::fNR, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fNR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fNR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fMR (obj_t v, location_type l)
      {
        return symbol_type (token::fMR, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fMR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fMR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fIRF (obj_t v, location_type l)
      {
        return symbol_type (token::fIRF, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fIRF (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fIRF, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fRESSEIZES (obj_t v, location_type l)
      {
        return symbol_type (token::fRESSEIZES, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fRESSEIZES (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fRESSEIZES, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fSTATE (obj_t v, location_type l)
      {
        return symbol_type (token::fSTATE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fSTATE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fSTATE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fSETSUM (obj_t v, location_type l)
      {
        return symbol_type (token::fSETSUM, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fSETSUM (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fSETSUM, v, l);
```

```
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fRESUTIL (obj_t v, location_type l)
      {
        return symbol_type (token::fRESUTIL, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fRESUTIL (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fRESUTIL, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_QUEUE (obj_t v, location_type l)
      {
        return symbol_type (token::QUEUE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_QUEUE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::QUEUE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fNQ (obj_t v, location_type l)
      {
        return symbol_type (token::fNQ, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fNQ (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fNQ, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fFIRSTINQ (obj_t v, location_type l)
      {
        return symbol_type (token::fFIRSTINQ, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fFIRSTINQ (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fFIRSTINQ, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fLASTINQ (obj_t v, location_type l)
      {
        return symbol_type (token::fLASTINQ, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fLASTINQ (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fLASTINQ, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fSAQUE (obj_t v, location_type l)
      {
        return symbol_type (token::fSAQUE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
```

```
      make_fSAQUE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fSAQUE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fAQUE (obj_t v, location_type l)
      {
        return symbol_type (token::fAQUE, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fAQUE (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fAQUE, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fENTATRANK (obj_t v, location_type l)
      {
        return symbol_type (token::fENTATRANK, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fENTATRANK (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fENTATRANK, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_SET (obj_t v, location_type l)
      {
        return symbol_type (token::SET, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_SET (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::SET, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fNUMSET (obj_t v, location_type l)
      {
        return symbol_type (token::fNUMSET, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fNUMSET (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fNUMSET, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_VARI (obj_t v, location_type l)
      {
        return symbol_type (token::VARI, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_VARI (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::VARI, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_FORM (obj_t v, location_type l)
      {
        return symbol_type (token::FORM, std::move (v), std::move (l));
      }
```

```
#else
      static
      symbol_type
      make_FORM (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::FORM, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fNUMGR (obj_t v, location_type l)
      {
        return symbol_type (token::fNUMGR, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fNUMGR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fNUMGR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_fATRGR (obj_t v, location_type l)
      {
        return symbol_type (token::fATRGR, std::move (v), std::move (l));
      }
#else
      static
      symbol_type
      make_fATRGR (const obj_t& v, const location_type& l)
      {
        return symbol_type (token::fATRGR, v, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_LPAREN (location_type l)
      {
        return symbol_type (token::LPAREN, std::move (l));
      }
#else
      static
      symbol_type
      make_LPAREN (const location_type& l)
      {
        return symbol_type (token::LPAREN, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_RPAREN (location_type l)
      {
        return symbol_type (token::RPAREN, std::move (l));
      }
#else
      static
      symbol_type
      make_RPAREN (const location_type& l)
      {
        return symbol_type (token::RPAREN, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_LBRACKET (location_type l)
      {
        return symbol_type (token::LBRACKET, std::move (l));
      }
#else
      static
      symbol_type
      make_LBRACKET (const location_type& l)
      {
        return symbol_type (token::LBRACKET, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_RBRACKET (location_type l)
```

```
      {
        return symbol_type (token::RBRACKET, std::move (l));
      }
#else
      static
      symbol_type
      make_RBRACKET (const location_type& l)
      {
        return symbol_type (token::RBRACKET, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_PLUS (location_type l)
      {
        return symbol_type (token::PLUS, std::move (l));
      }
#else
      static
      symbol_type
      make_PLUS (const location_type& l)
      {
        return symbol_type (token::PLUS, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_MINUS (location_type l)
      {
        return symbol_type (token::MINUS, std::move (l));
      }
#else
      static
      symbol_type
      make_MINUS (const location_type& l)
      {
        return symbol_type (token::MINUS, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_STAR (location_type l)
      {
        return symbol_type (token::STAR, std::move (l));
      }
#else
      static
      symbol_type
      make_STAR (const location_type& l)
      {
        return symbol_type (token::STAR, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_POWER (location_type l)
      {
        return symbol_type (token::POWER, std::move (l));
      }
#else
      static
      symbol_type
      make_POWER (const location_type& l)
      {
        return symbol_type (token::POWER, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_SLASH (location_type l)
      {
        return symbol_type (token::SLASH, std::move (l));
      }
#else
      static
      symbol_type
      make_SLASH (const location_type& l)
      {
        return symbol_type (token::SLASH, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
```

```
      static
      symbol_type
      make_LESS (location_type l)
      {
        return symbol_type (token::LESS, std::move (l));
      }
#else
      static
      symbol_type
      make_LESS (const location_type& l)
      {
        return symbol_type (token::LESS, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_GREATER (location_type l)
      {
        return symbol_type (token::GREATER, std::move (l));
      }
#else
      static
      symbol_type
      make_GREATER (const location_type& l)
      {
        return symbol_type (token::GREATER, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_ASSIGN (location_type l)
      {
        return symbol_type (token::ASSIGN, std::move (l));
      }
#else
      static
      symbol_type
      make_ASSIGN (const location_type& l)
      {
        return symbol_type (token::ASSIGN, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_COMMA (location_type l)
      {
        return symbol_type (token::COMMA, std::move (l));
      }
#else
      static
      symbol_type
      make_COMMA (const location_type& l)
      {
        return symbol_type (token::COMMA, l);
      }
#endif
#if 201103L <= YY_CPLUSPLUS
      static
      symbol_type
      make_NEG (location_type l)
      {
        return symbol_type (token::NEG, std::move (l));
      }
#else
      static
      symbol_type
      make_NEG (const location_type& l)
      {
        return symbol_type (token::NEG, l);
      }
#endif


    class context
    {
    public:
      context (const genesyspp_parser& yyparser, const symbol_type& yyla);
      const symbol_type& lookahead () const YY_NOEXCEPT { return yyla_; }
      symbol_kind_type token () const YY_NOEXCEPT { return yyla_.kind (); }
      const location_type& location () const YY_NOEXCEPT { return yyla_.location; }

      int expected_tokens (symbol_kind_type yyarg[], int yyargn) const;

    private:
```

```
      const genesyspp_parser& yyparser_;
      const symbol_type& yyla_;
    };

  private:
#if YY_CPLUSPLUS < 201103L
    genesyspp_parser (const genesyspp_parser&);
    genesyspp_parser& operator= (const genesyspp_parser&);
#endif


    typedef short state_type;

    int yy_syntax_error_arguments_ (const context& yyctx,
                                    symbol_kind_type yyarg[], int yyargn) const;

    virtual std::string yysyntax_error_ (const context& yyctx) const;
    static state_type yy_lr_goto_state_ (state_type yystate, int yysym);

    static bool yy_pact_value_is_default_ (int yyvalue);

    static bool yy_table_value_is_error_ (int yyvalue);

    static const short yypact_ninf_;
    static const signed char yytable_ninf_;

    static symbol_kind_type yytranslate_ (int t);

    static std::string yytnamerr_ (const char *yystr);

    static const char* const yytname_[];


    // Tables.
    // YYPACT[STATE-NUM] -- Index in YYTABLE of the portion describing
    // STATE-NUM.
    static const short yypact_[];

    // YYDEFACT[STATE-NUM] -- Default reduction number in state STATE-NUM.
    // Performed when YYTABLE does not specify something else to do.  Zero
    // means the default is an error.
    static const signed char yydefact_[];

    // YYPGOTO[NTERM-NUM].
    static const short yypgoto_[];

    // YYDEFGOTO[NTERM-NUM].
    static const unsigned char yydefgoto_[];

    // YYTABLE[YYPACT[STATE-NUM]] -- What to do in state STATE-NUM.  If
    // positive, shift that token.  If negative, reduce the rule whose
    // number is the opposite.  If YYTABLE_NINF, syntax error.
    static const short yytable_[];

    static const short yycheck_[];

    // YYSTOS[STATE-NUM] -- The (internal number of the) accessing
    // symbol of state STATE-NUM.
    static const signed char yystos_[];

    // YYR1[YYN] -- Symbol number of symbol that rule YYN derives.
    static const signed char yyr1_[];

    // YYR2[YYN] -- Number of symbols on the right hand side of rule YYN.
    static const signed char yyr2_[];


#if YYDEBUG
    // YYRLINE[YYN] -- Source line where rule number YYN was defined.
    static const short yyrline_[];
    virtual void yy_reduce_print_ (int r) const;
    virtual void yy_stack_print_ () const;

    int yydebug_;
    std::ostream* yycdebug_;

    template <typename Base>
    void yy_print_ (std::ostream& yyo, const basic_symbol<Base>& yysym) const;
#endif

    template <typename Base>
    void yy_destroy_ (const char* yymsg, basic_symbol<Base>& yysym) const;

  private:
    struct by_state
    {
      by_state () YY_NOEXCEPT;
```

```
      typedef state_type kind_type;

      by_state (kind_type s) YY_NOEXCEPT;

      by_state (const by_state& that) YY_NOEXCEPT;

      void clear () YY_NOEXCEPT;

      void move (by_state& that);

      symbol_kind_type kind () const YY_NOEXCEPT;

      enum { empty_state = 0 };

      state_type state;
    };

    struct stack_symbol_type : basic_symbol<by_state>
    {
      typedef basic_symbol<by_state> super_type;
      stack_symbol_type ();
      stack_symbol_type (YY_RVREF (stack_symbol_type) that);
      stack_symbol_type (state_type s, YY_MOVE_REF (symbol_type) sym);
#if YY_CPLUSPLUS < 201103L
      stack_symbol_type& operator= (stack_symbol_type& that);

      stack_symbol_type& operator= (const stack_symbol_type& that);
#endif
    };

    template <typename T, typename S = std::vector<T> >
    class stack
    {
    public:
      // Hide our reversed order.
      typedef typename S::iterator iterator;
      typedef typename S::const_iterator const_iterator;
      typedef typename S::size_type size_type;
      typedef typename std::ptrdiff_t index_type;

      stack (size_type n = 200)
        : seq_ (n)
      {}

#if 201103L <= YY_CPLUSPLUS
      stack (const stack&) = delete;
      stack& operator= (const stack&) = delete;
#endif

      const T&
      operator[] (index_type i) const
      {
        return seq_[size_type (size () - 1 - i)];
      }

      T&
      operator[] (index_type i)
      {
        return seq_[size_type (size () - 1 - i)];
      }

      void
      push (YY_MOVE_REF (T) t)
      {
        seq_.push_back (T ());
        operator[] (0).move (t);
      }

      void
      pop (std::ptrdiff_t n = 1) YY_NOEXCEPT
      {
        for (; 0 < n; --n)
          seq_.pop_back ();
      }

      void
      clear () YY_NOEXCEPT
      {
        seq_.clear ();
      }

      index_type
      size () const YY_NOEXCEPT
      {
        return index_type (seq_.size ());
      }
```

```

      const_iterator
      begin () const YY_NOEXCEPT
      {
        return seq_.begin ();
      }

      const_iterator
      end () const YY_NOEXCEPT
      {
        return seq_.end ();
      }

      class slice
      {
      public:
        slice (const stack& stack, index_type range)
          : stack_ (stack)
          , range_ (range)
        {}

        const T&
        operator[] (index_type i) const
        {
          return stack_[range_ - i];
        }

      private:
        const stack& stack_;
        index_type range_;
      };

    private:
#if YY_CPLUSPLUS < 201103L
      stack (const stack&);
      stack& operator= (const stack&);
#endif
      S seq_;
    };


    typedef stack<stack_symbol_type> stack_type;

    stack_type yystack_;

    void yypush_ (const char* m, YY_MOVE_REF (stack_symbol_type) sym);

    void yypush_ (const char* m, state_type s, YY_MOVE_REF (symbol_type) sym);

    void yypop_ (int n = 1);

    enum
    {
      yylast_ = 1776,
      yynnts_ = 22,
      yyfinal_ = 119
    };


    // User arguments.
    genesyspp_driver& driver;

  };

  inline
  genesyspp_parser::symbol_kind_type
  genesyspp_parser::yytranslate_ (int t)
  {
    // YYTRANSLATE[TOKEN-NUM] -- Symbol number corresponding to
    // TOKEN-NUM as returned by yylex.
    static
    const signed char
    translate_table[] =
    {
       0,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      88,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
       2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
```

```
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     2,     2,     2,     2,
      2,     2,     2,     2,     2,     2,     1,     2,     3,     4,
      5,     6,     7,     8,     9,    10,    11,    12,    13,    14,
     15,    16,    17,    18,    19,    20,    21,    22,    23,    24,
     25,    26,    27,    28,    29,    30,    31,    32,    33,    34,
     35,    36,    37,    38,    39,    40,    41,    42,    43,    44,
     45,    46,    47,    48,    49,    50,    51,    52,    53,    54,
     55,    56,    57,    58,    59,    60,    61,    62,    63,    64,
     65,    66,    67,    68,    69,    70,    71,    72,    73,    74,
     75,    76,    77,    78,    79,    80,    81,    82,    83,    84,
     85,    86,    87,    89
  };
  // Last valid token kind.
  const int code_max = 343;

  if (t <= 0)
    return symbol_kind::S_YYEOF;
  else if (t <= code_max)
    return YY_CAST (symbol_kind_type, translate_table[t]);
  else
    return symbol_kind::S_YYUNDEF;
}

// basic_symbol.
template <typename Base>
genesyspp_parser::basic_symbol<Base>::basic_symbol (const basic_symbol& that)
  : Base (that)
  , value ()
  , location (that.location)
{
  switch (this->kind ())
  {
    case symbol_kind::S_NUMD: // NUMD
    case symbol_kind::S_NUMH: // NUMH
    case symbol_kind::S_CTEZERO: // CTEZERO
    case symbol_kind::S_oLE: // oLE
    case symbol_kind::S_oGE: // oGE
    case symbol_kind::S_oEQ: // oEQ
    case symbol_kind::S_oNE: // oNE
    case symbol_kind::S_oAND: // oAND
    case symbol_kind::S_oOR: // oOR
    case symbol_kind::S_oNAND: // oNAND
    case symbol_kind::S_oXOR: // oXOR
    case symbol_kind::S_oNOT: // oNOT
    case symbol_kind::S_fSIN: // fSIN
    case symbol_kind::S_fCOS: // fCOS
    case symbol_kind::S_fROUND: // fROUND
    case symbol_kind::S_fMOD: // fMOD
    case symbol_kind::S_fTRUNC: // fTRUNC
    case symbol_kind::S_fFRAC: // fFRAC
    case symbol_kind::S_fEXP: // fEXP
    case symbol_kind::S_fSQRT: // fSQRT
    case symbol_kind::S_fLOG: // fLOG
    case symbol_kind::S_fLN: // fLN
    case symbol_kind::S_fVAL: // fVAL
    case symbol_kind::S_fEVAL: // fEVAL
    case symbol_kind::S_fLENG: // fLENG
    case symbol_kind::S_fRND1: // fRND1
    case symbol_kind::S_fEXPO: // fEXPO
    case symbol_kind::S_fNORM: // fNORM
    case symbol_kind::S_fUNIF: // fUNIF
    case symbol_kind::S_fWEIB: // fWEIB
    case symbol_kind::S_fLOGN: // fLOGN
    case symbol_kind::S_fGAMM: // fGAMM
    case symbol_kind::S_fERLA: // fERLA
    case symbol_kind::S_fTRIA: // fTRIA
    case symbol_kind::S_fBETA: // fBETA
    case symbol_kind::S_fDISC: // fDISC
    case symbol_kind::S_fTNOW: // fTNOW
    case symbol_kind::S_fTFIN: // fTFIN
    case symbol_kind::S_fMAXREP: // fMAXREP
    case symbol_kind::S_fNUMREP: // fNUMREP
    case symbol_kind::S_fIDENT: // fIDENT
    case symbol_kind::S_cIF: // cIF
    case symbol_kind::S_cELSE: // cELSE
    case symbol_kind::S_cFOR: // cFOR
```

```
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.copy< obj_t > (YY_MOVE (that.value));
        break;

      default:
        break;
    }

  }



  template <typename Base>
  genesyspp_parser::symbol_kind_type
  genesyspp_parser::basic_symbol<Base>::type_get () const YY_NOEXCEPT
  {
    return this->kind ();
  }

  template <typename Base>
  bool
  genesyspp_parser::basic_symbol<Base>::empty () const YY_NOEXCEPT
  {
    return this->kind () == symbol_kind::S_YYEMPTY;
  }

  template <typename Base>
  void
  genesyspp_parser::basic_symbol<Base>::move (basic_symbol& s)
  {
    super_type::move (s);
    switch (this->kind ())
    {
      case symbol_kind::S_NUMD: // NUMD
      case symbol_kind::S_NUMH: // NUMH
      case symbol_kind::S_CTEZERO: // CTEZERO
      case symbol_kind::S_oLE: // oLE
      case symbol_kind::S_oGE: // oGE
      case symbol_kind::S_oEQ: // oEQ
      case symbol_kind::S_oNE: // oNE
```

```
      case symbol_kind::S_oAND: // oAND
      case symbol_kind::S_oOR: // oOR
      case symbol_kind::S_oNAND: // oNAND
      case symbol_kind::S_oXOR: // oXOR
      case symbol_kind::S_oNOT: // oNOT
      case symbol_kind::S_fSIN: // fSIN
      case symbol_kind::S_fCOS: // fCOS
      case symbol_kind::S_fROUND: // fROUND
      case symbol_kind::S_fMOD: // fMOD
      case symbol_kind::S_fTRUNC: // fTRUNC
      case symbol_kind::S_fFRAC: // fFRAC
      case symbol_kind::S_fEXP: // fEXP
      case symbol_kind::S_fSQRT: // fSQRT
      case symbol_kind::S_fLOG: // fLOG
      case symbol_kind::S_fLN: // fLN
      case symbol_kind::S_fVAL: // fVAL
      case symbol_kind::S_fEVAL: // fEVAL
      case symbol_kind::S_fLENG: // fLENG
      case symbol_kind::S_fRND1: // fRND1
      case symbol_kind::S_fEXPO: // fEXPO
      case symbol_kind::S_fNORM: // fNORM
      case symbol_kind::S_fUNIF: // fUNIF
      case symbol_kind::S_fWEIB: // fWEIB
      case symbol_kind::S_fLOGN: // fLOGN
      case symbol_kind::S_fGAMM: // fGAMM
      case symbol_kind::S_fERLA: // fERLA
      case symbol_kind::S_fTRIA: // fTRIA
      case symbol_kind::S_fBETA: // fBETA
      case symbol_kind::S_fDISC: // fDISC
      case symbol_kind::S_fTNOW: // fTNOW
      case symbol_kind::S_fTFIN: // fTFIN
      case symbol_kind::S_fMAXREP: // fMAXREP
      case symbol_kind::S_fNUMREP: // fNUMREP
      case symbol_kind::S_fIDENT: // fIDENT
      case symbol_kind::S_cIF: // cIF
      case symbol_kind::S_cELSE: // cELSE
      case symbol_kind::S_cFOR: // cFOR
      case symbol_kind::S_cTO: // cTO
      case symbol_kind::S_cDO: // cDO
      case symbol_kind::S_ATRIB: // ATRIB
      case symbol_kind::S_CSTAT: // CSTAT
      case symbol_kind::S_fTAVG: // fTAVG
      case symbol_kind::S_ILLEGAL: // ILLEGAL
      case symbol_kind::S_RESOURCE: // RESOURCE
      case symbol_kind::S_fNR: // fNR
      case symbol_kind::S_fMR: // fMR
      case symbol_kind::S_fIRF: // fIRF
      case symbol_kind::S_fRESSEIZES: // fRESSEIZES
      case symbol_kind::S_fSTATE: // fSTATE
      case symbol_kind::S_fSETSUM: // fSETSUM
      case symbol_kind::S_fRESUTIL: // fRESUTIL
      case symbol_kind::S_QUEUE: // QUEUE
      case symbol_kind::S_fNQ: // fNQ
      case symbol_kind::S_fFIRSTINQ: // fFIRSTINQ
      case symbol_kind::S_fLASTINQ: // fLASTINQ
      case symbol_kind::S_fSAQUE: // fSAQUE
      case symbol_kind::S_fAQUE: // fAQUE
      case symbol_kind::S_fENTATRANK: // fENTATRANK
      case symbol_kind::S_SET: // SET
      case symbol_kind::S_fNUMSET: // fNUMSET
      case symbol_kind::S_VARI: // VARI
      case symbol_kind::S_FORM: // FORM
      case symbol_kind::S_fNUMGR: // fNUMGR
      case symbol_kind::S_fATRGR: // fATRGR
      case symbol_kind::S_input: // input
      case symbol_kind::S_expression: // expression
      case symbol_kind::S_numero: // numero
      case symbol_kind::S_aritmetica: // aritmetica
      case symbol_kind::S_logica: // logica
      case symbol_kind::S_relacional: // relacional
      case symbol_kind::S_comando: // comando
      case symbol_kind::S_comandoIF: // comandoIF
      case symbol_kind::S_comandoFOR: // comandoFOR
      case symbol_kind::S_function: // function
      case symbol_kind::S_kernelFunction: // kernelFunction
      case symbol_kind::S_trigonFunction: // trigonFunction
      case symbol_kind::S_mathFunction: // mathFunction
      case symbol_kind::S_probFunction: // probFunction
      case symbol_kind::S_userFunction: // userFunction
      case symbol_kind::S_listaparm: // listaparm
      case symbol_kind::S_atributo: // atributo
      case symbol_kind::S_variavel: // variavel
      case symbol_kind::S_formula: // formula
      case symbol_kind::S_atribuicao: // atribuicao
      case symbol_kind::S_pluginFunction: // pluginFunction
        value.move< obj_t > (YY_MOVE (s.value));
        break;
```

```
      default:
        break;
    }

    location = YY_MOVE (s.location);
  }

  // by_kind.
  inline
  genesyspp_parser::by_kind::by_kind ()
    : kind_ (symbol_kind::S_YYEMPTY)
  {}

#if 201103L <= YY_CPLUSPLUS
  inline
  genesyspp_parser::by_kind::by_kind (by_kind&& that)
    : kind_ (that.kind_)
  {
    that.clear ();
  }
#endif

  inline
  genesyspp_parser::by_kind::by_kind (const by_kind& that)
    : kind_ (that.kind_)
  {}

  inline
  genesyspp_parser::by_kind::by_kind (token_kind_type t)
    : kind_ (yytranslate_ (t))
  {}

  inline
  void
  genesyspp_parser::by_kind::clear () YY_NOEXCEPT
  {
    kind_ = symbol_kind::S_YYEMPTY;
  }

  inline
  void
  genesyspp_parser::by_kind::move (by_kind& that)
  {
    kind_ = that.kind_;
    that.clear ();
  }

  inline
  genesyspp_parser::symbol_kind_type
  genesyspp_parser::by_kind::kind () const YY_NOEXCEPT
  {
    return kind_;
  }

  inline
  genesyspp_parser::symbol_kind_type
  genesyspp_parser::by_kind::type_get () const YY_NOEXCEPT
  {
    return this->kind ();
  }

} // yy
#line 3241 "../GenesysParser.h"



#endif // !YY_YY_GENESYSPARSER_H_INCLUDED
```

8.241 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/location.hh File Reference

```
#include <iostream>
#include <string>
```

Include dependency graph for location.hh:

This graph shows which files directly or indirectly include this file:

Classes

- class yy::position

  A point in a source file.
- class yy::location

  Two points in a source file.

Namespaces

- yy

Macros

- #define YY_NULLPTR ((void∗)0)

Functions

- position & yy::operator+= (position &res, position::counter_type width)

  Add width columns, in place.
- position yy::operator+ (position res, position::counter_type width)

  Add width columns.
- position & yy::operator-= (position &res, position::counter_type width)

  Subtract width columns, in place.
- position yy::operator- (position res, position::counter_type width)

  Subtract width columns.
- template<typename YYChar >
  std::basic_ostream< YYChar > & yy::operator<< (std::basic_ostream< YYChar > &ostr, const position &pos)

  Intercept output stream redirection.
- location & yy::operator+= (location &res, const location &end)

  Join two locations, in place.
- location yy::operator+ (location res, const location &end)

  Join two locations.
- location & yy::operator+= (location &res, location::counter_type width)

  Add width columns to the end position, in place.
- location yy::operator+ (location res, location::counter_type width)

  Add width columns to the end position.
- location & yy::operator-= (location &res, location::counter_type width)

  Subtract width columns to the end position, in place.
- location yy::operator- (location res, location::counter_type width)

  Subtract width columns to the end position.
- template<typename YYChar >
  std::basic_ostream< YYChar > & yy::operator<< (std::basic_ostream< YYChar > &ostr, const location &loc)

  Intercept output stream redirection.

8.241.1 Detailed Description

Define the yy::location class.

Definition in file location.hh.

8.241.2 Macro Definition Documentation

8.241.2.1 YY_NULLPTR `#define YY_NULLPTR ((void*)0)`

Definition at line 52 of file location.hh.

8.242 location.hh

```
00001 // A Bison parser, made by GNU Bison 3.7.6.
00002
00003 // Locations for Bison parsers in C++
00004
00005 // Copyright (C) 2002-2015, 2018-2021 Free Software Foundation, Inc.
00006
00007 // This program is free software: you can redistribute it and/or modify
00008 // it under the terms of the GNU General Public License as published by
00009 // the Free Software Foundation, either version 3 of the License, or
00010 // (at your option) any later version.
00011
00012 // This program is distributed in the hope that it will be useful,
00013 // but WITHOUT ANY WARRANTY; without even the implied warranty of
00014 // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
00015 // GNU General Public License for more details.
00016
00017 // You should have received a copy of the GNU General Public License
00018 // along with this program.  If not, see <https://www.gnu.org/licenses/>.
00019
00020 // As a special exception, you may create a larger work that contains
00021 // part or all of the Bison parser skeleton and distribute that work
00022 // under terms of your choice, so long as that work isn't itself a
00023 // parser generator using the skeleton or a modified version thereof
00024 // as a parser skeleton.  Alternatively, if you modify or redistribute
00025 // the parser skeleton itself, you may (at your option) remove this
00026 // special exception, which will cause the skeleton and the resulting
00027 // Bison output files to be licensed under the GNU General Public
00028 // License without this special exception.
00029
00030 // This special exception was added by the Free Software Foundation in
00031 // version 2.2 of Bison.
00032
00038 #ifndef YY_YY_LOCATION_HH_INCLUDED
00039 # define YY_YY_LOCATION_HH_INCLUDED
00040
00041 # include <iostream>
00042 # include <string>
00043
00044 # ifndef YY_NULLPTR
00045 #  if defined __cplusplus
00046 #   if 201103L <= __cplusplus
00047 #    define YY_NULLPTR nullptr
00048 #   else
00049 #    define YY_NULLPTR 0
00050 #   endif
00051 #  else
00052 #   define YY_NULLPTR ((void*)0)
00053 #  endif
00054 # endif
00055
00056 namespace yy {
00057 #line 58 "../location.hh"
00058
00060   class position
00061   {
00062   public:
00064     typedef const std::string filename_type;
00066     typedef int counter_type;
00067
00069     explicit position (filename_type* f = YY_NULLPTR,
00070                        counter_type l = 1,
00071                        counter_type c = 1)
00072       : filename (f)
00073       , line (l)
00074       , column (c)
00075     {}
00076
00077
00079     void initialize (filename_type* fn = YY_NULLPTR,
00080                      counter_type l = 1,
00081                      counter_type c = 1)
00082     {
00083       filename = fn;
00084       line = l;
00085       column = c;
00086     }
00087
00091     void lines (counter_type count = 1)
00092     {
00093       if (count)
00094         {
00095           column = 1;
00096           line = add_ (line, count, 1);
00097         }
00098     }
```

```
00099
00101     void columns (counter_type count = 1)
00102     {
00103       column = add_ (column, count, 1);
00104     }
00108     filename_type* filename;
00110     counter_type line;
00112     counter_type column;
00113
00114   private:
00116     static counter_type add_ (counter_type lhs, counter_type rhs, counter_type min)
00117     {
00118       return lhs + rhs < min ? min : lhs + rhs;
00119     }
00120   };
00121
00123   inline position&
00124   operator+= (position& res, position::counter_type width)
00125   {
00126     res.columns (width);
00127     return res;
00128   }
00129
00131   inline position
00132   operator+ (position res, position::counter_type width)
00133   {
00134     return res += width;
00135   }
00136
00138   inline position&
00139   operator-= (position& res, position::counter_type width)
00140   {
00141     return res += -width;
00142   }
00143
00145   inline position
00146   operator- (position res, position::counter_type width)
00147   {
00148     return res -= width;
00149   }
00150
00155   template <typename YYChar>
00156   std::basic_ostream<YYChar>&
00157   operator« (std::basic_ostream<YYChar>& ostr, const position& pos)
00158   {
00159     if (pos.filename)
00160       ostr « *pos.filename « ':';
00161     return ostr « pos.line « '.' « pos.column;
00162   }
00163
00165   class location
00166   {
00167   public:
00169     typedef position::filename_type filename_type;
00171     typedef position::counter_type counter_type;
00172
00174     location (const position& b, const position& e)
00175       : begin (b)
00176       , end (e)
00177     {}
00178
00180     explicit location (const position& p = position ())
00181       : begin (p)
00182       , end (p)
00183     {}
00184
00186     explicit location (filename_type* f,
00187                        counter_type l = 1,
00188                        counter_type c = 1)
00189       : begin (f, l, c)
00190       , end (f, l, c)
00191     {}
00192
00193
00195     void initialize (filename_type* f = YY_NULLPTR,
00196                      counter_type l = 1,
00197                      counter_type c = 1)
00198     {
00199       begin.initialize (f, l, c);
00200       end = begin;
00201     }
00202
00205   public:
00207     void step ()
00208     {
00209       begin = end;
00210     }
```

```
    void columns (counter_type count = 1)
    {
      end += count;
    }

    void lines (counter_type count = 1)
    {
      end.lines (count);
    }
  public:
    position begin;
    position end;
  };

  inline location&
  operator+= (location& res, const location& end)
  {
    res.end = end.end;
    return res;
  }

  inline location
  operator+ (location res, const location& end)
  {
    return res += end;
  }

  inline location&
  operator+= (location& res, location::counter_type width)
  {
    res.columns (width);
    return res;
  }

  inline location
  operator+ (location res, location::counter_type width)
  {
    return res += width;
  }

  inline location&
  operator-= (location& res, location::counter_type width)
  {
    return res += -width;
  }

  inline location
  operator- (location res, location::counter_type width)
  {
    return res -= width;
  }

  template <typename YYChar>
  std::basic_ostream<YYChar>&
  operator« (std::basic_ostream<YYChar>& ostr, const location& loc)
  {
    location::counter_type end_col
      = 0 < loc.end.column ? loc.end.column - 1 : 0;
    ostr « loc.begin;
    if (loc.end.filename
        && (!loc.begin.filename
            || *loc.begin.filename != *loc.end.filename))
      ostr « '-' « loc.end.filename « ':' « loc.end.line « '.' « end_col;
    else if (loc.begin.line < loc.end.line)
      ostr « '-' « loc.end.line « '.' « end_col;
    else if (loc.begin.column < end_col)
      ostr « '-' « end_col;
    return ostr;
  }

} // yy
#line 303 "../location.hh"

#endif // !YY_YY_LOCATION_HH_INCLUDED
```

8.243 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/obj_t.cpp File Reference

```
#include "obj_t.h"
```

#include <string>

Include dependency graph for obj_t.cpp:

8.244 obj_t.cpp

```
#include "obj_t.h"
#include <string>

obj_t::obj_t() {

}

obj_t::~obj_t() {

}

obj_t::obj_t(double v, std::string t) {
    valor = v;
    tipo = t;
    id = 0;
}

obj_t::obj_t(double v, std::string t, unsigned long uid) {
    valor = v;
    tipo = t;
    id = uid;
}
```

8.245 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBisonFlex/obj↩ _t.h File Reference

#include <string>

Include dependency graph for obj_t.h:

This graph shows which files directly or indirectly include this file:

Classes

- class obj_t

8.246 obj_t.h

```
#ifndef OBJ_T_H
#define OBJ_T_H
#include <string>

class obj_t {
public:
    obj_t();
    virtual ~obj_t();
    obj_t(double v, std::string t);
    obj_t(double v, std::string t, unsigned long uid);
    double valor;
    std::string tipo;
    unsigned long id;
};

#endif
```

8.247 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBison↩ Flex/parser/bisonparser.yy File Reference

8.248 bisonparser.yy

```
%skeleton "lalr1.cc" /* -*- C++ -*- */
%require "3.0.4"
%defines
```

```
%define api.parser.class {genesyspp_parser} //Name of the parses class
%define api.token.constructor //let that way or change YY_DECL prototype
%define api.value.type variant
%define parse.assert //Checks for constructor and destructor(?)
%code requires
{
    #include <string>
    #include <cmath>
    #include "obj_t.h"
    #include "../../util/Util.h"
    #include "../Attribute.h"

    /****begin_Includes_plugins****/

        /**begin_Includes:Variable**/
        #include "../../../plugins/elements/Variable.h"
        /**end_Includes:Variable**/

        /**begin_Includes:Queue**/
        #include "../../../plugins/elements/Queue.h"
        /**end_Includes:Queue**/

        /**begin_Includes:Formula**/
        #include "../../../plugins/elements/Formula.h"
        /**end_Includes:Formula**/

        /**begin_Includes:Resource**/
        #include "../../../plugins/elements/Resource.h"
        /**end_Includes:Resource**/

        /**begin_Includes:Set**/
        #include "../../../plugins/elements/Set.h"
        /**end_Includes:Set**/

    /****end_Includes_plugins****/

    class genesyspp_driver;
}

//%define api.value.type {obj_t} //c++ types for semantic values
 //c++ types for semantic values
// The parsing context.
%param { genesyspp_driver& driver } //aditional params for yylex/yyparse
%locations // allows for more accurate syntax error messages.
%initial-action
{
  // Initialize the initial location.
  //@$.begin.filename = @$.end.filename = &driver.getFile();
};
 //for debugging
%define parse.error verbose //error level to show
%code
{
# include "Genesys++-driver.h"

}

// numbers
%token <obj_t> NUMD
%token <obj_t> NUMH
%token <obj_t> CTEZERO
// relational operators
%token <obj_t> oLE
%token <obj_t> oGE
%token <obj_t> oEQ
%token <obj_t> oNE
// logic operators
%token <obj_t> oAND
%token <obj_t> oOR
%token <obj_t> oNAND
%token <obj_t> oXOR
%token <obj_t> oNOT
// trigonometric functions
%token <obj_t> fSIN
%token <obj_t> fCOS
// math functions
%token <obj_t> fROUND
%token <obj_t> fMOD
%token <obj_t> fTRUNC
%token <obj_t> fFRAC
%token <obj_t> fEXP
%token <obj_t> fSQRT
%token <obj_t> fLOG
%token <obj_t> fLN
// string functions
%token <obj_t> fVAL
%token <obj_t> fEVAL
```

```
%token <obj_t> fLENG
// probability distributionsformulaValue
%token <obj_t> fRND1
%token <obj_t> fEXPO
%token <obj_t> fNORM
%token <obj_t> fUNIF
%token <obj_t> fWEIB
%token <obj_t> fLOGN
%token <obj_t> fGAMM
%token <obj_t> fERLA
%token <obj_t> fTRIA
%token <obj_t> fBETA
%token <obj_t> fDISC
// simulation infos
%token <obj_t> fTNOW
%token <obj_t> fTFIN
%token <obj_t> fMAXREP
%token <obj_t> fNUMREP
%token <obj_t> fIDENT
// algoritmic functions
%token <obj_t> cIF
%token <obj_t> cELSE
%token <obj_t> cFOR
%token <obj_t> cTO
%token <obj_t> cDO
// kernel elements
%token <obj_t> ATRIB
%token <obj_t> CSTAT
%token <obj_t> fTAVG
%token <obj_t> ILLEGAL     /* illegal token */

/****begin_Tokens_plugins****/

    /**begin_Tokens:Resource**/
    %token <obj_t> RESOURCE
    %token <obj_t> fNR
    %token <obj_t> fMR
    %token <obj_t> fIRF
    %token <obj_t> fRESSEIZES
    %token <obj_t> fSTATE
    %token <obj_t> fSETSUM
    %token <obj_t> fRESUTIL
    /**end_Tokens:Resource**/

    /**begin_Tokens:Queue**/
    %token <obj_t> QUEUE
    %token <obj_t> fNQ
    %token <obj_t> fFIRSTINQ
    %token <obj_t> fLASTINQ
    %token <obj_t> fSAQUE
    %token <obj_t> fAQUE
    %token <obj_t> fENTATRANK
    /**end_Tokens:Queue**/

    /**begin_Tokens:Set**/
    %token <obj_t> SET
    %token <obj_t> fNUMSET
    /**end_Tokens:Set**/

    /**begin_Tokens:Variable**/
    %token <obj_t> VARI
    /**end_Tokens:Variable**/

    /**begin_Tokens:Formula**/
    %token <obj_t> FORM
    /**end_Tokens:Formula**/

    /**begin_Tokens:EntityGroup**/
    %token <obj_t> fNUMGR
    %token <obj_t> fATRGR
    /**end_Tokens:EntityGroup**/

/****end_Tokens_plugins****/

%token LPAREN "("
%token RPAREN ")"
%token LBRACKET "["
%token RBRACKET "]"
%token PLUS "+"
%token MINUS "-"
%token STAR "*"
%token POWER "^"
%token SLASH "/"
%token LESS "<"
%token GREATER ">"
%token ASSIGN "="
%token COMMA ","
```

```
%token END 0 "end of file" //need to declare, as bison doesnt in especific situation

%type <obj_t> input
%type <obj_t> expression
%type <obj_t> aritmetica
%type <obj_t> logica
%type <obj_t> relacional
%type <obj_t> comando
%type <obj_t> comandoIF
%type <obj_t> comandoFOR
%type <obj_t> function
%type <obj_t> numero
%type <obj_t> atributo
%type <obj_t> atribuicao
%type <obj_t> kernelFunction
%type <obj_t> trigonFunction
%type <obj_t> mathFunction
%type <obj_t> probFunction
%type <obj_t> pluginFunction
%type <obj_t> userFunction
%type <obj_t> listaparm
%type <obj_t> illegal

/****begin_TypeObj_plugins****/

    /**begin_TypeObj::Variable**/
    %type <obj_t> variavel
    /**end_TypeObj::Variable**/

    /**begin_TypeObj::Formula**/
    %type <obj_t> formula
    /**end_TypeObj::Formula**/

/****end_TypeObj_plugins****/

%left oAND oOR;
%left oNOT;
%left oLE oGE oEQ oNE LESS GREATER LBRACKET cELSE;
%left MINUS PLUS;
%left STAR SLASH;
%precedence NEG;
%left fROUND fMOD fTRUNC fFRAC fLOG fLN fSQRT;

//%printer { yyoutput « $$; } <*>; //prints when something
%%

input:
      expression    { driver.setResult($1.valor);}
    | error '\n'        { yyerrok; }
    ;

expression:
      numero                          {$$.valor = $1.valor;}
    | function                        {$$.valor = $1.valor;}
    | comando                         {$$.valor = $1.valor;}
    | atribuicao                      {$$.valor = $1.valor;}
    | aritmetica                      {$$.valor = $1.valor;}
    | logica                          {$$.valor = $1.valor;}
    | relacional                      {$$.valor = $1.valor;}
    | "(" expression ")"              {$$.valor = $2.valor;}
    | atributo                        {$$.valor = $1.valor;}

/****begin_Expression_plugins****/

    /**begin_Expression:Variable**/
        | variavel                        {$$.valor = $1.valor;}
    /**end_Expression:Variable**/

    /**begin_Expression:Formula**/
        | formula                         {$$.valor = $1.valor;}
    /**end_Expression:Formula**/

/****end_Expression_plugins****/
    ;

numero:
     NUMD     { $$.valor = $1.valor;}
    | NUMH    { $$.valor = $1.valor;}
    ;

aritmetica:
     expression PLUS expression      { $$.valor = $1.valor + $3.valor;}
    | expression MINUS expression    { $$.valor = $1.valor - $3.valor;}
    | expression SLASH expression    { $$.valor = $1.valor / $3.valor;}
    | expression STAR expression     { $$.valor = $1.valor * $3.valor;}
    | expression POWER expression    { $$.valor = pow($1.valor,$3.valor);}
    | MINUS expression %prec NEG     { $$.valor = -$2.valor;}
```

```
    ;

logica:
      expression oAND expression    { $$.valor = (int) $1.valor && (int) $3.valor;}
    | expression oOR  expression    { $$.valor = (int) $1.valor || (int) $3.valor;}
    | expression oNAND expression   { $$.valor = !((int) $1.valor && (int) $3.valor);}
    | expression oXOR  expression   { $$.valor = (!(int) $1.valor && (int) $3.valor) || ((int) $1.valor && !(int) $3.valor);}
    | oNOT expression               { $$.valor = !(int) $2.valor;}
    ;

relacional:
      expression LESS  expression        { $$.valor = $1.valor < $3.valor ? 1 : 0;}
    | expression GREATER expression      { $$.valor = $1.valor > $3.valor ? 1 : 0;}
    | expression oLE  expression         { $$.valor = $1.valor <= $3.valor ? 1 : 0;}
    | expression oGE  expression         { $$.valor = $1.valor >= $3.valor ? 1 : 0;}
    | expression oEQ  expression         { $$.valor = $1.valor == $3.valor ? 1 : 0;}
    | expression oNE  expression         { $$.valor = $1.valor != $3.valor ? 1 : 0;}
    ;

comando:
      comandoIF     { $$.valor = $1.valor; }
    | comandoFOR    { $$.valor = $1.valor; }
    ;

comandoIF:
      cIF expression expression cELSE expression   { $$.valor = $2.valor != 0 ? $3.valor : $5.valor; }
    | cIF expression expression                   { $$.valor = $2.valor != 0 ? $3.valor : 0;}
    ;

// \todo: check for function/need, for now will let cout (these should be commands for program, not expression
comandoFOR:
     cFOR variavel "=" expression cTO expression cDO atribuicao  {$$.valor = 0; }
    | cFOR atributo "=" expression cTO expression cDO atribuicao  {$$.valor = 0; }
    ;

function:
     mathFunction        { $$.valor = $1.valor; }
    | trigonFunction     { $$.valor = $1.valor; }
    | probFunction       { $$.valor = $1.valor; }
    | kernelFunction     { $$.valor = $1.valor; }
    | pluginFunction     { $$.valor = $1.valor; }
    | userFunction       { $$.valor = $1.valor; }
    ;

kernelFunction:
      fTNOW      { $$.valor = driver.getModel()->getSimulation()->getSimulatedTime();}
    | fTFIN      { $$.valor = driver.getModel()->getSimulation()->getReplicationLength();}
    | CSTAT      { $$.valor = 0; }
    | fTAVG  "(" CSTAT ")"     {
                    StatisticsCollector* cstat = ((StatisticsCollector*)(driver.getModel()->getElements()->getElement(Util::TypeOf<StatisticsCollector>(), $3.id)));
                    double value = cstat->getStatistics()->average();
                    $$.valor = value; }
    ;

trigonFunction:
      fSIN   "(" expression ")"   { $$.valor = sin($3.valor); }
    | fCOS   "(" expression ")"   { $$.valor = cos($3.valor); }
    ;

mathFunction:
      fROUND "(" expression ")"     { $$.valor = round($3.valor);}
    | fFRAC  "(" expression ")"     { $$.valor = $3.valor - (int) $3.valor;}
    | fTRUNC "(" expression ")"     { $$.valor = trunc($3.valor);}
    | fEXP "(" expression ")"       { $$.valor = exp($3.valor);}
    | fSQRT "(" expression ")"      { $$.valor = sqrt($3.valor);}
    | fLOG "(" expression ")"       { $$.valor = log10($3.valor);}
    | fLN "(" expression ")"        { $$.valor = log($3.valor);}
    | fMOD   "(" expression "," expression ")" { $$.valor = (int) $3.valor % (int) $5.valor; }
    ;

probFunction:
      fRND1                     { $$.valor = driver.sampler()->sampleUniform(0.0,1.0);}
    | fEXPO  "(" expression ")"  { $$.valor = driver.sampler()->sampleExponential($3.valor);}
    | fNORM  "(" expression "," expression ")"  { $$.valor = driver.sampler()->sampleNormal($3.valor,$5.valor);}
    | fUNIF  "(" expression "," expression ")"  { $$.valor = driver.sampler()->sampleUniform($3.valor,$5.valor);}
    | fWEIB  "(" expression "," expression ")"  { $$.valor = driver.sampler()->sampleWeibull($3.valor,$5.valor);}
    | fLOGN  "(" expression "," expression ")"  { $$.valor = driver.sampler()->sampleLogNormal($3.valor,$5.valor);}
    | fGAMM  "(" expression "," expression ")"  { $$.valor = driver.sampler()->sampleGamma($3.valor,$5.valor);}
```

```
    | fERLA  "(" expression "," expression ")"  { $$.valor = driver.sampler()->sampleErlang($3.valor,$5.valor);}
    | fTRIA  "(" expression "," expression "," expression ")"   { $$.valor = driver.sampler()->sampleTriangular($3.valor,$5.valor,$7.valor);}
    | fBETA  "(" expression "," expression "," expression "," expression ")"  { $$.valor = driver.sampler()->sampleBeta($3.valor,$5.valor,$7.valor,$9.valor);}
    | fDISC  "(" listaparm ")"                  { $$.valor = driver.sampler()->sampleDiscrete(0,0); /*@TODO: NOT IMPLEMENTED YET*/ }
    ;


//Maybe user defined functions, check if continues on the parser, for now returns the value of expression
userFunction:
      "USER" "(" expression ")"         { $$.valor = $3.valor; }
    ;

//Probably returns parameters for something, check if continues on the parser, for now does nothing
listaparm:
      listaparm "," expression "," expression    {/*@TODO: NOT IMPLEMENTED YET*/}
    | expression "," expression                  {/*@TODO: NOT IMPLEMENTED YET*/}
    ;

//If illegal token, verifies if throws exception or set error message
illegal:
    ILLEGAL           {
        driver.setResult(-1);
        if(driver.getThrowsException()){
            if($1.valor == 0){
              throw std::string("Literal nao encontrado");
            }else if($1.valor == 1){
              throw std::string("Caracter invalido encontrado");
            }
        } else {
            if($1.valor == 0){
              driver.setErrorMessage(std::string("Literal nao encontrado"));
            }else if($1.valor == 1){
                driver.setErrorMessage(std::string("Caracter invalido encontrado"));
            }
        }
    }
    ;


// 20181003  ATRIB now returns the attribute ID not the attribute value anymore. So, now get the attribute value for the current entity
atributo:
    ATRIB      {
        double attributeValue = 0.0;
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue($1.id);
        }
        //std::cout « "Passei" « std::endl;
        $$.valor = attributeValue;
    }
    | ATRIB LBRACKET expression RBRACKET  {
        double attributeValue = 0.0;
        std::string index = std::to_string(static_cast<unsigned int>($3.valor));
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(index, $1.id);
        }
        $$.valor = attributeValue;
    }
    | ATRIB LBRACKET expression "," expression RBRACKET  {
        double attributeValue = 0.0;
        std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor));
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(index, $1.id);
        }
        $$.valor = attributeValue;
    }
    | ATRIB LBRACKET expression "," expression "," expression RBRACKET  {
        double attributeValue = 0.0;
        std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor))+","+std::to_string(static_cast<unsigned int>($7.valor));
        if (driver.getModel()->getSimulation()->getCurrentEntity() != nullptr) {
```

```
            // it could crach because there may be no current entity, if the parse is running before simulation and therefore there is no CurrentEntity
            attributeValue = driver.getModel()->getSimulation()->getCurrentEntity()->getAttributeValue(index, $1.id);
        }
        $$.valor = attributeValue;
    }
    ;

/****begin_ExpressionProdution_plugins****/

    /**begin_ExpressionProdution:Variable**/
    variavel    : VARI  {$$.valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->value();}
                | VARI LBRACKET expression RBRACKET     {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor));
                    $$.valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->value(index); }
                | VARI LBRACKET expression "," expression RBRACKET      {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor));
                    $$.valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->value(index);}
                | VARI LBRACKET expression "," expression "," expression RBRACKET    {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor))+","+std::to_string(static_cast<unsigned int>($7.valor));
                    $$.valor = ((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->value(index);}
                ;
    /**end_ExpressionProdution:Variable**/

    /**begin_ExpressionProdution:Formula**/
    // \todo: THERE IS A PROBLEM WITH FORMULA: TO EVALUATE THE FORMULA EXPRESSION, PARSER IS REINVOKED, AND THEN IT CRASHES (NO REENTRACE?)
    formula     : FORM      {
                    $$.valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), $1.id)))->value();}
                | FORM LBRACKET expression RBRACKET {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor));
                    $$.valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), $1.id)))->value(index);}
                | FORM LBRACKET expression "," expression RBRACKET {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor));
                    $$.valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), $1.id)))->value(index);}
                | FORM LBRACKET expression "," expression "," expression RBRACKET {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor))+","+std::to_string(static_cast<unsigned int>($7.valor));
                    $$.valor = ((Formula*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), $1.id)))->value(index);}
                ;
    /**end_ExpressionProdution:Formula**/
    /****end_ExpressionProdution_plugins****/

    //Check if want to set the atributo or variavel with expression or just return the expression value, for now just returns expression value
    atribuicao  : ATRIB ASSIGN expression    {
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue("", $1.id, $3.valor);
                    $$.valor = $3.valor; }
                | ATRIB LBRACKET expression RBRACKET ASSIGN expression    {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor));
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue(index, $1.id, $6.valor);
                    $$.valor = $6.valor; }
                | ATRIB LBRACKET expression "," expression RBRACKET ASSIGN expression   {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor));
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue(index, $1.id, $8.valor);
                    $$.valor = $8.valor;}
                | ATRIB LBRACKET expression "," expression "," expression RBRACKET ASSIGN expression    {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor))+","+std::to_string(static_cast<unsigned int>($7.valor));
```

```
                    driver.getModel()->getSimulation()->getCurrentEntity()->setAttributeValue(index, $1.id, $10.valor);
                    $$.valor = $10.valor; }
    /****begin_Assignment_plugins****/
    /**begin_Assignment:Variable**/
                | VARI ASSIGN expression        {

((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->setValue($3.valor);
                    $$.valor = $3.valor; }
                | VARI LBRACKET expression RBRACKET ASSIGN expression    {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor));

((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->setValue(index, $6.valor);
                    $$.valor = $6.valor; }
                | VARI LBRACKET expression "," expression RBRACKET ASSIGN expression   {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor));

((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->setValue(index, $8.valor);
                    $$.valor = $8.valor; }
                | VARI LBRACKET expression "," expression "," expression RBRACKET ASSIGN expression   {
                    std::string index = std::to_string(static_cast<unsigned int>($3.valor))+","+std::to_string(static_cast<unsigned int>($5.valor))+","+std::to_string(static_cast<unsigned int>($7.valor));

((Variable*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), $1.id)))->setValue(index, $10.valor);
                    $$.valor = $10.valor; }
    /**end_Assignment:Variable**/

/****end_Assignment_plugins****/
            ;


pluginFunction  :
      CTEZERO                                       { $$.valor = 0; }
/**begin_FunctionProdution_plugins**/
/**begin_FunctionProdution:Queue**/
    | fNQ       "(" QUEUE ")"       {   //std::cout « "Queue ID: " « $3.id « ", Size: " « ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->size() « std::endl;
                                        $$.valor = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->size();}
    | fLASTINQ  "(" QUEUE ")"       {/*For now does nothing because need acces to list of QUEUE, or at least the last element*/ }
    | fFIRSTINQ "(" QUEUE ")"       {
                if (((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->size() > 0){
                    //id da 1a entidade da fila, talvez pegar nome
                    $$.valor = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->first()->getEntity()->getId();
                }else{
                    $$.valor = 0;
                } }
    | fSAQUE "(" QUEUE "," ATRIB ")"   {
                Util::identification queueID = $3.id;
                Util::identification attrID = $5.id;
                double sum = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->sumAttributesFromWaiting(attrID);
                $$.valor = sum; }
    | fAQUE "(" QUEUE "," NUMD "," ATRIB ")" {
                Util::identification queueID = $3.id;
                Util::identification attrID = $7.id;
                double value = ((Queue*)(driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), $3.id)))->getAttributeFromWaitingRank($5.valor-1, attrID); // rank starts on 0 in genesys
                $$.valor = value; }
/**end_FunctionProdution:Queue**/

/**begin_FunctionProdution:Resource**/
    | fMR       "(" RESOURCE ")"   { $$.valor = ((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), $3.id))->getCapacity();}
    | fNR       "(" RESOURCE ")"        { $$.valor = ((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), $3.id))->getNumberBusy();}
    | fRESSEIZES "(" RESOURCE ")"       { /*For now does nothing because needs get Seizes, check with teacher*/}
    | fSTATE    "(" RESOURCE ")"        {  $$.valor = static_cast<int>(((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), $3.id))->getResourceState()); }
```

```
    | fIRF       "(" RESOURCE ")"        { $$.valor = ((Resource*)driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), $3.id))->getResourceState() == Resource::ResourceState::FAILED ? 1 : 0; }
    | fSETSUM    "(" SET ")"  {
                unsigned int count=0;
                Resource* res;
                List<ModelElement*>* setList = ((Set*)driver.getModel()->getElements()->getElement(Util::TypeOf<Set>(),$3.id))->getElementSet();
                for (std::list<ModelElement*>::iterator it = setList->list()->begin(); it!=setList->list()->end(); it++) {
                    res = dynamic_cast<Resource*>(*it);
                    if (res != nullptr) {
                        if (res->getResourceState()==Resource::ResourceState::BUSY) {
                            count++;
                        }
                    }
                }
                $$.valor = count; }
/**end_FunctionProdution:Resource**/

/**begin_FunctionProdution:Set**/
    | fNUMSET    "(" SET ")"    { $$.valor = ((Set*)driver.getModel()->getElements()->getElement(Util::TypeOf<Set>(),$3.id))->getElementSet()->size(); }

/**end_FunctionProdution:Set**/
/****end_FunctionProdution_plugins****/
    ;


%%
void
yy::genesyspp_parser::error (const location_type& l,
                          const std::string& m)
{
  driver.error (l, m);
}
```

8.249 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBison↩ Flex/parser/lexerparser.ll File Reference

8.250 lexerparser.ll

```
%{ /* -*- C++ -*- */
# include <cerrno>
# include <climits>
# include <cstdlib>
# include <string>
# include <locale>
# include <list>
# include "Genesys++-driver.h"
# include "GenesysParser.h"
# include "obj_t.h"
# include "../../util/Util.h"
# include "../../util/List.h"
# include "../ModelElement.h"
# include "../Attribute.h"
# include "../StatisticsCollector.h"
/**begin_Includes_plugins**/
/**begin_Includes:Variable**/
#include "../../../plugins/elements/Variable.h"
/**end_Includes:Variable**/
/**begin_Includes:Queue**/
#include "../../../plugins/elements/Queue.h"
/**end_Includes:Queue**/
/**begin_Includes:Formula**/
#include "../../../plugins/elements/Formula.h"
/**end_Includes:Formula**/
/**begin_Includes:Resource**/
#include "../../../plugins/elements/Resource.h"
/**end_Includes:Resource**/
/**begin_Includes:Set**/
#include "../../../plugins/elements/Set.h"
/**end_Includes:Set**/
/**end_Includes_plugins**/


// Work around an incompatibility in flex (at least versions
// 2.5.31 through 2.5.33): it generates code that does
// not conform to C89.  See Debian bug 333231
```

```
// <http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=333231>.
# undef yywrap
# define yywrap() 1

// The location of the current token.
static yy::location loc;
%}
%option noyywrap nounput batch noinput

%{
  // Code run each time a pattern is matched.
  # define YY_USER_ACTION  loc.columns (yyleng);
%}

I     ~[A-Za-z]+
DD     [0-9]+([eE][-]?[0-9]+)?
RR     [0-9]+[.][0-9]+([eE][-]?[0-9]+)?
L      [A-Za-z0-9_.]+

%%

%{
  // Code run each time yylex is called.
  loc.step ();
%}


[0][xX][aA-fF0-9]+  {
    //Hexadecimal number
    //Will not fail because of regex
    //std::string text("Found Hexadecimal: ");
    //text += yytext;
    //driver.getModel()->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, text);
    return yy::genesyspp_parser::make_NUMH(obj_t(atof(yytext), std::string("Hexadecimal")),loc);
      }

{RR}  {
       //Float number
       //Will not fail because of regex
       //std::string text("Found Float: ");
       //text += yytext;
       //driver.getModel()->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, text);
       return yy::genesyspp_parser::make_NUMD(obj_t(atof(yytext),std::string("Float")), loc);
      }

{DD}  {
       //Decimal number
       //Will not fail because of regex
       //std::string text("Found Decimal: ");
       //text += yytext;
       //driver.getModel()->getTraceManager()->trace(Util::TraceLevel::L9_mostDetailed, text);
       return yy::genesyspp_parser::make_NUMD(obj_t(atof(yytext),std::string("Decimal")), loc);
      }

%{// special symbols %}
[<][=]   { return (yy::genesyspp_parser::make_oLE(obj_t(0, std::string(yytext)), loc));}
[>][=]   { return (yy::genesyspp_parser::make_oGE(obj_t(0, std::string(yytext)), loc));}
[=][=]   { return (yy::genesyspp_parser::make_oEQ(obj_t(0, std::string(yytext)), loc));}
[<][>]   { return (yy::genesyspp_parser::make_oNE(obj_t(0, std::string(yytext)), loc));}
[<] {return yy::genesyspp_parser::make_LESS(loc);}
[>] {return yy::genesyspp_parser::make_GREATER(loc);}
[(] {return yy::genesyspp_parser::make_LPAREN(loc);}
[)] {return yy::genesyspp_parser::make_RPAREN(loc);}
[+] {return yy::genesyspp_parser::make_PLUS(loc);}
[-] {return yy::genesyspp_parser::make_MINUS(loc);}
[*] {return yy::genesyspp_parser::make_STAR(loc);}
[\^] {return yy::genesyspp_parser::make_POWER(loc);}
[/] {return yy::genesyspp_parser::make_SLASH(loc);}
[=] {return yy::genesyspp_parser::make_ASSIGN(loc);}
[,] {return yy::genesyspp_parser::make_COMMA(loc);}
[\[] {return yy::genesyspp_parser::make_LBRACKET(loc);}
[\]] {return yy::genesyspp_parser::make_RBRACKET(loc);}

%{// boolean values %}
[tT][rR][uU][eE]      {return yy::genesyspp_parser::make_NUMD(obj_t(1, std::string(yytext)), loc);}
[fF][aA][lL][sS][eE]  {return yy::genesyspp_parser::make_NUMD(obj_t(0, std::string(yytext)), loc);}

%{// algorithmic functions %}
[iI][fF]              {return yy::genesyspp_parser::make_cIF(obj_t(0, std::string(yytext)), loc);}
[eE][lL][sS][eE]      {return yy::genesyspp_parser::make_cELSE(obj_t(0, std::string(yytext)), loc);}
[fF][oO][rR]          {return yy::genesyspp_parser::make_cFOR(obj_t(0, std::string(yytext)), loc);}
[tT][oO]              {return yy::genesyspp_parser::make_cTO(obj_t(0, std::string(yytext)), loc);}
[dD][oO]              {return yy::genesyspp_parser::make_cDO(obj_t(0, std::string(yytext)), loc);}

%{// logic funcions %}
[aA][nN][dD]    {return yy::genesyspp_parser::make_oAND(obj_t(0, std::string(yytext)), loc);}
[oO][rR]        {return yy::genesyspp_parser::make_oOR(obj_t(0, std::string(yytext)), loc);}
```

```
[nN][aA][nN][dD]    {return yy::genesyspp_parser::make_oNAND(obj_t(0, std::string(yytext)), loc);}
[xX][oO][rR]        {return yy::genesyspp_parser::make_oXOR(obj_t(0, std::string(yytext)), loc);}
[nN][oO][tT]    {return yy::genesyspp_parser::make_oNOT(obj_t(0, std::string(yytext)), loc);}

%{// trigonometric functions %}
[sS][iI][nN]      {return yy::genesyspp_parser::make_fSIN(obj_t(0, std::string(yytext)), loc);}
[cC][oO][sS]      {return yy::genesyspp_parser::make_fCOS(obj_t(0, std::string(yytext)), loc);}

%{// math funcions %}
[rR][oO][uU][nN][dD]  {return yy::genesyspp_parser::make_fROUND(obj_t(0, std::string(yytext)), loc);}
[mM][oO][dD]          {return yy::genesyspp_parser::make_fMOD(obj_t(0, std::string(yytext)), loc);}
[tT][rR][uU][nN][cC]  {return yy::genesyspp_parser::make_fTRUNC(obj_t(0, std::string(yytext)), loc);}
[fF][rR][aA][cC]      {return yy::genesyspp_parser::make_fFRAC(obj_t(0, std::string(yytext)), loc);}
[eE][xX][pP]          {return yy::genesyspp_parser::make_fEXP(obj_t(0, std::string(yytext)), loc);}
[sS][qQ][rR][tT]      {return yy::genesyspp_parser::make_fSQRT(obj_t(0, std::string(yytext)), loc);}
[lL][oO][gG]          {return yy::genesyspp_parser::make_fLOG(obj_t(0, std::string(yytext)), loc);}
[lL][nN]              {return yy::genesyspp_parser::make_fLN(obj_t(0, std::string(yytext)), loc);}

%{// string functions %}
[vV][aA][lL]          {return yy::genesyspp_parser::make_fVAL(obj_t(0, std::string(yytext)), loc);}
[eE][vV][aA][lL]      {return yy::genesyspp_parser::make_fEVAL(obj_t(0, std::string(yytext)), loc);}
[lL][eE][nN][gG]      {return yy::genesyspp_parser::make_fLENG(obj_t(0, std::string(yytext)), loc);}

%{// probability distributions %}
[rR][nN][dD]      {return yy::genesyspp_parser::make_fRND1(obj_t(0, std::string(yytext)), loc);}
[eE][xX][pP][oO]  {return yy::genesyspp_parser::make_fEXPO(obj_t(0, std::string(yytext)), loc);}
[nN][oO][rR][mM]  {return yy::genesyspp_parser::make_fNORM(obj_t(0, std::string(yytext)), loc);}
[uU][nN][iI][fF]  {return yy::genesyspp_parser::make_fUNIF(obj_t(0, std::string(yytext)), loc);}
[wW][eE][iI][bB]  {return yy::genesyspp_parser::make_fWEIB(obj_t(0, std::string(yytext)), loc);}
[lL][oO][gG][nN]  {return yy::genesyspp_parser::make_fLOGN(obj_t(0, std::string(yytext)), loc);}
[gG][aA][mM][mM]  {return yy::genesyspp_parser::make_fGAMM(obj_t(0, std::string(yytext)), loc);}
[eE][rR][lL][aA]  {return yy::genesyspp_parser::make_fERLA(obj_t(0, std::string(yytext)), loc);}
[tT][rR][iI][aA]  {return yy::genesyspp_parser::make_fTRIA(obj_t(0, std::string(yytext)), loc);}
[bB][eE][tT][aA]  {return yy::genesyspp_parser::make_fBETA(obj_t(0, std::string(yytext)), loc);}
[dD][iI][sS][cC]  {return yy::genesyspp_parser::make_fDISC(obj_t(0, std::string(yytext)), loc);}

%{// simulation infos %}
[tT][nN][oO][wW]          {return yy::genesyspp_parser::make_fTNOW(obj_t(0, std::string(yytext)), loc);}
[tT][fF][iI][nN]          {return yy::genesyspp_parser::make_fTFIN(obj_t(0, std::string(yytext)), loc);}
[mM][aA][xX][rR][eE][pP]  {return yy::genesyspp_parser::make_fMAXREP(obj_t(0, std::string(yytext)), loc);}
[nN][uU][mM][rR][eE][pP]  {return yy::genesyspp_parser::make_fNUMREP(obj_t(0, std::string(yytext)), loc);}
[iI][dD][eE][nN][tT]      {return yy::genesyspp_parser::make_fIDENT(obj_t(0, std::string(yytext)), loc);}

[tT][aA][vV][gG]  {return yy::genesyspp_parser::make_fTAVG(obj_t(0, std::string(yytext)), loc);}


%{/****begin_Lexical_plugins****/%}

%{/**begin_Lexical:Resource**/%}
[nN][rR]                                    {return yy::genesyspp_parser::make_fNR(obj_t(0, std::string(yytext)), loc);}
[mM][rR]                                    {return yy::genesyspp_parser::make_fMR(obj_t(0, std::string(yytext)), loc);}
[iI][rR][fF]                                {return yy::genesyspp_parser::make_fIRF(obj_t(0, std::string(yytext)), loc);}
[sS][tT][aA][tT][eE]                        {return yy::genesyspp_parser::make_fSTATE(obj_t(0, std::string(yytext)), loc);}
[sS][eE][tT][sS][uU][mM]                    {return yy::genesyspp_parser::make_fSETSUM(obj_t(0, std::string(yytext)), loc);}
[rR][eE][sS][uU][tT][iI][lL]                {return yy::genesyspp_parser::make_fRESUTIL(obj_t(0, std::string(yytext)), loc);}
[rR][eE][sS][sS][eE][iI][zZ][eE][sS]        {return yy::genesyspp_parser::make_fRESSEIZES(obj_t(0, std::string(yytext)), loc);}
[iI][dD][lL][eE][_][rR][eE][sS]             {return yy::genesyspp_parser::make_NUMD(obj_t(-1, std::string(yytext)), loc);}
[bB][uU][sS][yY][_][rR][eE][sS]             {return yy::genesyspp_parser::make_NUMD(obj_t(-2, std::string(yytext)), loc);}
[iI][nN][aA][cC][tT][iI][vV][eE][_][rR][eE][sS] {return yy::genesyspp_parser::make_NUMD(obj_t(-3, std::string(yytext)), loc);}
[fF][aA][iI][lL][eE][dD][_][rR][eE][sS]     {return yy::genesyspp_parser::make_NUMD(obj_t(-4, std::string(yytext)), loc);}
%{/**end_Lexical:Resource**/%}

%{/**begin_Lexical:Queue**/%}
[nN][qQ]                          {return yy::genesyspp_parser::make_fNQ(obj_t(0, std::string(yytext)), loc);}
[lL][aA][sS][tT][iI][nN][qQ]      {return yy::genesyspp_parser::make_fLASTINQ(obj_t(0, std::string(yytext)), loc);}
[fF][iI][rR][sS][tT][iI][nN][qQ]  {return yy::genesyspp_parser::make_fFIRSTINQ(obj_t(0, std::string(yytext)), loc);}
[sS][aA][qQ][uU][eE]              {return yy::genesyspp_parser::make_fSAQUE(obj_t(0, std::string(yytext)), loc);}
```

```
[aA][qQ][uU][eE]                    {return yy::genesyspp_parser::make_fAQUE(obj_t(0, std::string(yytext)), loc);}
[eE][nN][tT][aA][tT][rR][aA][nN][kK] {return yy::genesyspp_parser::make_fENTATRANK(obj_t(0, std::string(yytext)), loc);}
%{/**end_Lexical:Queue**/%}

%{/**begin_Lexical:Set**/%}
[nN][uU][mM][sS][eE][tT]            {return yy::genesyspp_parser::make_fNUMSET(obj_t(0, std::string(yytext)), loc);}
%{/**end_Lexical:Set**/%}

%{/**begin_Lexical:EntityGroup**/%}
[nN][uU][mM][gG][rR]                {return yy::genesyspp_parser::make_fNUMGR(obj_t(0, std::string(yytext)), loc);}
[aA][tT][rR][gG][rR]                {return yy::genesyspp_parser::make_fATRGR(obj_t(0, std::string(yytext)), loc);}
%{/**end_Lexical:EntityGroup**/%}

%{/****end_Lexical_plugins****/%}


[ \t\n]        ;
T

{L}   {
        ModelElement* element;

        //std::cout « "Verificando o que é o Literal \""+std::string(yytext)+"\"\n";
        //std::cout « driver.getModel()->getInfos()->getName() « std::endl;
        // check if it is an ATTRIBUTE (and return the attribute ID (and not the value!)
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Attribute>(), std::string(yytext));
        if (element != nullptr) {
            //std::cout « "Found attribute" « std::string(yytext) « std::endl;
            return yy::genesyspp_parser::make_ATRIB(obj_t(0, Util::TypeOf<Attribute>(), element->getId()),loc);
        }

/****begin_LexicalLiterals_plugins****/

/**begin_LexicalLiterals:Variable**/
        // check VARIABLE
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Variable>(), std::string(yytext));
        if (element != nullptr) { // it is a variable
            Variable* var = static_cast<Variable*>(element);
            //double variableID = var->getId();// ->getValue(); // var->getId()
        //std::cout « "FOUND VARIABLE " « var->getName() «" ID " « var->getId() « std::endl;
            return yy::genesyspp_parser::make_VARI(obj_t(0, Util::TypeOf<Variable>(), var->getId()),loc);
        }
/**end_LexicalLiterals:Variable**/

/**begin_LexicalLiterals:Formula**/
        // check FORMULA
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Formula>(), std::string(yytext));
        if (element != nullptr) { // it is a FORMULA
            Formula* form = static_cast<Formula*>(element);
            //double formulaValue = form->getValue(); // return only formula ID
        //std::cout « "FOUND FORMULA " « form->getName() «" ID " « form->getId() « std::endl;
            return yy::genesyspp_parser::make_FORM(obj_t(0, Util::TypeOf<Formula>(), form->getId()),loc);
        }
/**end_LexicalLiterals:Formula**/

/**begin_LexicalLiterals:Queue**/
        // check QUEUE
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Queue>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_QUEUE(obj_t(0, Util::TypeOf<Queue>(), element->getId()),loc);
        }
/**end_LexicalLiterals:Queue**/

/**begin_LexicalLiterals:Resource**/
        // check RESOURCE
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Resource>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_RESOURCE(obj_t(0, Util::TypeOf<Resource>(), element->getId()),loc);
        }
/**end_LexicalLiterals:Resource**/

/**begin_LexicalLiterals:Set**/
```

```
        //check SET
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<Set>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_SET(obj_t(0, Util::TypeOf<Set>(), element->getId()),loc);
        }
/**end_LexicalLiterals:Set**/

/****end_LexicalLiterals_plugins****/

        //check CSTAT
        element = driver.getModel()->getElements()->getElement(Util::TypeOf<StatisticsCollector>(), std::string(yytext));
        if (element != nullptr) {
            return yy::genesyspp_parser::make_CSTAT(obj_t(0, Util::TypeOf<StatisticsCollector>(), element->getId()),loc);
        }

    // If no one before has identified this literal, then it is an ILLEGAL (not found, unknown) literal
        //Case not found retturns a illegal token
        return yy::genesyspp_parser::make_ILLEGAL(obj_t(0, std::string("Illegal")), loc);
      }

.       {return yy::genesyspp_parser::make_ILLEGAL(obj_t(1, std::string("Illegal")), loc);}

«EOF» {return yy::genesyspp_parser::make_END(loc);}


%%

void
genesyspp_driver::scan_begin_file ()
{
  //yy_flex_debug = trace_scanning;
  if (file.empty () || file == "-")
    yyin = stdin;
  else if (!(yyin = fopen (file.c_str (), "r")))
    {
      error ("cannot open " + file + ": " + strerror(errno));
      exit (EXIT_FAILURE);
    }
}

void genesyspp_driver::scan_begin_str ()
{
  //yy_flex_debug = trace_scanning;
  if(!str_to_parse.empty()){
    yy_scan_string (str_to_parse.c_str()); //maybe throw exception on else
  }else{
    std::string str("0");
    yy_scan_string (str.c_str()); //maybe throw exception on else
  }
}



void
genesyspp_driver::scan_end_file ()
{
  fclose (yyin);
}

void
genesyspp_driver::scan_end_str ()
{
  yy_delete_buffer(YY_CURRENT_BUFFER);
}
```

8.251 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBison↩ Flex/position.hh File Reference

```
#include "location.hh"
```

Include dependency graph for position.hh:

8.252 position.hh

```
// A Bison parser, made by GNU Bison 3.7.6.

// Starting with Bison 3.2, this file is useless: the structure it
// used to define is now defined in "location.hh".
//
// To get rid of this file:
// 1. add '%require "3.2"' (or newer) to your grammar file
// 2. remove references to this file from your build system
// 3. if you used to include it, include "location.hh" instead.

#include "location.hh"
```

8.253 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/parserBison↩ Flex/stack.hh File Reference

8.254 stack.hh

```
// A Bison parser, made by GNU Bison 3.7.6.

// Starting with Bison 3.2, this file is useless: the structure it
// used to define is now defined with the parser itself.
//
// To get rid of this file:
// 1. add '%require "3.2"' (or newer) to your grammar file
// 2. remove references to this file from your build system.
```

8.255 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserChanges↩ Information.cpp File Reference

```
#include "ParserChangesInformation.h"
```

Include dependency graph for ParserChangesInformation.cpp:

8.256 ParserChangesInformation.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserChangesInformation.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 20:42
 */

#include "ParserChangesInformation.h"

ParserChangesInformation::ParserChangesInformation() {
}

```

8.257 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserChanges↩ Information.h File Reference

```
#include <string>
#include <list>
```

Include dependency graph for ParserChangesInformation.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ParserChangesInformation

8.258 ParserChangesInformation.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserChangesInformation.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 20:42
 */

#ifndef PARSERCHANGESINFORMATION_H
#define PARSERCHANGESINFORMATION_H

#include <string>
#include <list>

class ParserChangesInformation {
public:
```

```
    ParserChangesInformation();
    virtual ~ParserChangesInformation() = default;

public: // gets and sets
private:
    std::string _includes = "";
    std::string _tokens = "";
    std::string _typeObjs = "";
    std::string _expressions;
    std::string _expressionProductions = "";
    std::string _assignments = "";
    std::string efunctionProdutions = "";

};

#endif /* PARSERCHANGESINFORMATION_H */

```

8.259 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserDefault↩ Impl1.cpp File Reference

`#include "ParserDefaultImpl1.h"`

Include dependency graph for ParserDefaultImpl1.cpp:

8.260 ParserDefaultImpl1.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserDefaultImpl1.cpp
```

```
 * Author: rafael.luiz.cancian
 *
 * Created on 2 de Agosto de 2018, 01:25
 */

#include "ParserDefaultImpl1.h"

//namespace GenesysKernel {

ParserDefaultImpl1::ParserDefaultImpl1(Model* model) {
    _model = model;
}

double ParserDefaultImpl1::parse(const std::string expression) { // may throw exception
    double result = std::atof(expression.c_str()); // change by a real parser
    return result;
}

std::string* ParserDefaultImpl1::getErrorMessage() {
    std::string* errorMsg = new std::string();
    return errorMsg; /* @ \todo: */
}

double ParserDefaultImpl1::parse(const std::string expression, bool* success, std::string*
 errorMessage) {
    try {
        double result = this->parse(expression);
        std::string temp(""); /* \todo: CHECK SCOPE OF VARIABLE */
        errorMessage = &temp;
        *success = true;
        return result;
    } catch (...) {
        std::string temp("Error parsing...");
        errorMessage = &temp;
        *success = false;
        return 0.0;
    }
}
//namespace\\}

```

8.261 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl1.h File Reference

`#include "Parser_if.h"`

Include dependency graph for ParserDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ParserDefaultImpl1

8.262 ParserDefaultImpl1.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   ParserDefaultImpl1.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 2 de Agosto de 2018, 01:25
00012  */
00013
00014 #ifndef PARSERDEFAULTIMPL1_H
00015 #define PARSERDEFAULTIMPL1_H
00016
00017 #include "Parser_if.h"
00018 //namespace GenesysKernel {
00019 class Model;
00020
00021 class ParserDefaultImpl1 : public Parser_if {
00022 public:
00023     ParserDefaultImpl1(Model* model);
00024     virtual ~ParserDefaultImpl1() = default;
00025 public:
00026     double parse(const std::string expression); // may throw exception
00027     double parse(const std::string expression, bool* success, std::string* errorMessage);
00028     std::string* getErrorMessage();
00029 private:
00030     Model* _model;
00031 };
00032 //namespace\\}
00033 #endif /* PARSERDEFAULTIMPL1_H */
00034
```

8.263 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserDefault↩ Impl2.cpp File Reference

`#include "ParserDefaultImpl2.h"`

Include dependency graph for ParserDefaultImpl2.cpp:

8.264 ParserDefaultImpl2.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserDefaultImpl2.cpp
 * Author: Joao Ortigara rafael.luiz.cancian (20181208-...(
 *
 *
 */

#include "ParserDefaultImpl2.h"

//using namespace GenesysKernel;

ParserDefaultImpl2::ParserDefaultImpl2(Model* model, Sampler_if* sampler, bool throws) {
    _model = model;
    _wrapper = genesyspp_driver(_model, sampler, throws);
}

double ParserDefaultImpl2::parse(const std::string expression) { // may throw exception
    _wrapper.setThrowsException(true);
    try {
        int res = _wrapper.parse_str(expression);
        if (res == 0) {
            return _wrapper.getResult();
        } else {
            throw std::string("Error parsing expression \"" + expression + "\"");
        }
    } catch (std::string e) {
        _model->getTracer()->trace(Util::TraceLevel::L1_errorFatal, e);
        return _wrapper.getResult();
    }
}

std::string* ParserDefaultImpl2::getErrorMessage() {
    std::string* errorMessage = new std::string(_wrapper.getErrorMessage());
    return errorMessage;
}

double ParserDefaultImpl2::parse(const std::string expression, bool* success, std::string*
 errorMessage) {
    _wrapper.setThrowsException(true); //false);
    int res = -1;
```

```
    try {
        res = _wrapper.parse_str(expression);
    } catch (...) {
        res = -1;
    }
    if (res == 0) {
        *success = true;
        return _wrapper.getResult();
    } else {
        *success = false;
        *errorMessage = _wrapper.getErrorMessage();
        return _wrapper.getResult();
    }
}

void ParserDefaultImpl2::setSampler(Sampler_if* _sampler) {
    _wrapper.setSampler(_sampler);
}

Sampler_if* ParserDefaultImpl2::sampler() const {
    return _wrapper.sampler();
}
```

8.265 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserDefaultImpl2.h File Reference

```
#include <string>
#include "Parser_if.h"
#include "Model.h"
#include "parserBisonFlex/Genesys++-driver.h"
```

Include dependency graph for ParserDefaultImpl2.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ParserDefaultImpl2

8.266 ParserDefaultImpl2.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserDefaultImpl2.h
 * Author: Joao Ortigara (20180801-20181207) rafael.luiz.cancian (20181208-...(
 *
 *
 */

#ifndef PARSERDEFAULTIMPL2_H
#define PARSERDEFAULTIMPL2_H

#include <string>
#include "Parser_if.h"
#include "Model.h"
#include "parserBisonFlex/Genesys++-driver.h"


//namespace GenesysKernel {

class ParserDefaultImpl2 : public Parser_if {
public:
    ParserDefaultImpl2(Model* model, Sampler_if* sampler, bool throws = false);
    virtual ~ParserDefaultImpl2() = default;
public:
    virtual double parse(const std::string expression); // may throw exception
    virtual double parse(const std::string expression, bool* success, std::string* errorMessage);
    virtual std::string* getErrorMessage();
    virtual void setSampler(Sampler_if* _sampler);
    virtual Sampler_if* sampler() const;
private:
    Model* _model;
    genesyspp_driver _wrapper;
};
//namespace\\}
#endif /* PARSERDEFAULTIMPL2_H */

```

8.267 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.cpp File Reference

```
#include "ParserManager.h"
```

Include dependency graph for ParserManager.cpp:

8.268 ParserManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserManager.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 20:34
 */

#include "ParserManager.h"

//using namespace GenesysKernel;

ParserManager::ParserManager() {
}


```

8.269 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ParserManager.h File Reference

```
#include "ParserChangesInformation.h"
```

Include dependency graph for ParserManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ParserManager
- struct ParserManager::NewParser
- struct ParserManager::GenerateNewParserResult

8.270 ParserManager.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ParserManager.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 20:34
 */

#ifndef PARSERMANAGER_H
#define PARSERMANAGER_H

#include "ParserChangesInformation.h"

//namespace GenesysKernel {

class ParserManager {
public:

    struct NewParser {
        std::string bisonFilename;
```

```
        std::string flexFilename;
        std::string compiledParserFilename;
    };

    struct GenerateNewParserResult {
        bool result;
        std::string bisonMessages;
        std::string lexMessages;
        std::string compilationMessages;
        NewParser newParser;
    };
public:
    ParserManager();
    virtual ~ParserManager() = default;
public:
    ParserManager::GenerateNewParserResult generateNewParser(ParserChangesInformation* changes);
    bool connectNewParser(ParserManager::NewParser newParser);
private:
};
//namespace\\}
#endif /* PARSERMANAGER_H */

```

8.271 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PersistentObject_↩ base.h File Reference

```
#include <map>
#include <string>
```

Include dependency graph for PersistentObject_base.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PersistentObject_base

8.272 PersistentObject_base.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   persistentObject_base.h
 * Author: rlcancian
 *
 * Created on 9 de abril de 2021, 15:26
 */

#ifndef PERSISTENTOBJECT_BASE_H
#define PERSISTENTOBJECT_BASE_H

#include <map>
#include <string>

class PersistentObject_base {
public:
    PersistentObject_base() {
    }
    virtual ~PersistentObject_base() = default;
protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields) = 0;
    virtual std::map<std::string, std::string>* _saveInstance() = 0;
};

#endif /* PERSISTENTOBJECT_BASE_H */

```

8.273 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Plugin.cpp File Reference

```
#include "Plugin.h"
#include "Model.h"
#include "SourceModelComponent.h"
#include "SinkModelComponent.h"
#include <assert.h>
```

Include dependency graph for Plugin.cpp:

8.274 Plugin.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
/*
 * File:   Plugin.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 12:58
 */

#include "Plugin.h"
#include "Model.h"
#include "SourceModelComponent.h"
#include "SinkModelComponent.h"
//#include "Assign.h"
#include <assert.h>

//using namespace GenesysKernel;

Plugin::Plugin(StaticGetPluginInformation getInformation) {
    this->_StatMethodGetInformation = getInformation;
    try {
        PluginInformation* infos = _StatMethodGetInformation();
        this->_pluginInfo = infos;
        this->_isValidPlugin = true;
    } catch (...) {
        this->_isValidPlugin = false;
    }
}

PluginInformation* Plugin::getPluginInfo() const {
    return _pluginInfo;
}

bool Plugin::isIsValidPlugin() const {
    return _isValidPlugin;
}

//Plugin::Plugin(std::string pluginTypename, bool source, bool drain) {
//    //_fullfilename = fullfilename;
//    _source = source;
//    _drain = drain;
//}

ModelElement* Plugin::loadNew(Model* model, std::map<std::string, std::string>* fields) {
    if (this->_pluginInfo->isComponent()) {
        return _loadNewComponent(model, fields);
    } else {
        return _loadNewElement(model, fields);
    }
}

bool Plugin::loadAndInsertNew(Model* model, std::map<std::string, std::string>* fields) {
    if (this->_pluginInfo->isComponent()) {
        ModelComponent* newComp = _loadNewComponent(model, fields);
        if (newComp != nullptr) {
            //model->getTraceManager()->trace(newComp->show());
            return true; //model->components()->insert(newComp);
        }
    } else {
        ModelElement* newElem = _loadNewElement(model, fields);
        if (newElem != nullptr) {
            //model->getTraceManager()->trace(newElem->show());
            return true; //model->elements()->insert(this->_pluginInfo->pluginTypename(), newElem);
        }
    }
    return false;
}

ModelComponent* Plugin::_loadNewComponent(Model* model, std::map<std::string, std::string>* fields) {
    //return this->_pluginInfo->loader(model, fields);
    StaticLoaderComponentInstance loader = this->_pluginInfo->GetComponentLoader();
    ModelComponent* newElementOrComponent = loader(model, fields);
    return newElementOrComponent;
}

ModelElement* Plugin::_loadNewElement(Model* model, std::map<std::string, std::string>* fields) {
    StaticLoaderElementInstance loader = this->_pluginInfo->getElementLoader();
    ModelElement* newElementOrComponent = loader(model, fields);
    return newElementOrComponent;
}
```

8.275 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Plugin.h File Reference

```
#include <string>
#include <functional>
#include <list>
#include "../util/Util.h"
#include "PluginInformation.h"
```

Include dependency graph for Plugin.h:

Classes

- class Plugin

8.276 Plugin.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Plugin.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 12:58
00012  */
00013
00014 #ifndef PLUGIN_H
00015 #define PLUGIN_H
00016
00017 #include <string>
00018 #include <functional>
00019 #include <list>
00020 #include "../util/Util.h"
00021
00022 #include "PluginInformation.h"
00023
00024 //namespace GenesysKernel {
00025
00030 class Plugin {
00031 public:
00032     Plugin(std::string filename_so_dll);
00033     Plugin(StaticGetPluginInformation getInformation); // temporary. Just while compiled together
00034     virtual ~Plugin() = default;
00035 public:
00036     bool isIsValidPlugin() const;
00037     PluginInformation* getPluginInfo() const;
00038 public:
00039     ModelElement* loadNew(Model* model, std::map<std::string, std::string>* fields);
00040     bool loadAndInsertNew(Model* model, std::map<std::string, std::string>* fields);
00041 private:
00042     ModelComponent* _loadNewComponent(Model* model, std::map<std::string, std::string>* fields);
00043     ModelElement* _loadNewElement(Model* model, std::map<std::string, std::string>* fields);
00044 private: // read only
00045     bool _isValidPlugin;
00046     PluginInformation* _pluginInfo;
00047 private:
00048     StaticGetPluginInformation _StatMethodGetInformation;
00049 };
00050 //namespace\\}
00051 #endif /* PLUGIN_H */
00052
```

8.277 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginConnector_if.h File Reference

```
#include <string>
#include "Plugin.h"
```

Include dependency graph for PluginConnector_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PluginConnector_if

8.278 PluginConnector_if.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   PluginConnector_if.h
00009  * Author: rlcancian
00010  *
00011  * Created on 9 de Setembro de 2019, 19:17
00012  */
00013
00014 #ifndef PLUGINCONNECTOR_IF_H
00015 #define PLUGINCONNECTOR_IF_H
00016
00017 #include<string>
00018 #include "Plugin.h"
00019
00020 //namespace GenesysKernel {
00021
00022 class PluginConnector_if {
00023 public:
00024     virtual Plugin* check(const std::string dynamicLibraryFilename) = 0;
00025     virtual Plugin* connect(const std::string dynamicLibraryFilename) = 0;
00026     virtual bool disconnect(const std::string dynamicLibraryFilename) = 0;
00027     virtual bool disconnect(Plugin* plugin) = 0;
00028 };
00029 //namespace\\}
00030 #endif /* PLUGINCONNECTOR_IF_H */
00031
```

8.279 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginConnectorDummyImpl1.cpp File Reference

```
#include "PluginConnectorDummyImpl1.h"
#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Process.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Dummy.h"
#include "../../plugins/components/Route.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Leave.h"
#include "../../plugins/components/Write.h"
#include "../../plugins/components/Hold.h"
#include "../../plugins/components/DropOff.h"
#include "../../plugins/components/Batch.h"
#include "../../plugins/components/Separate.h"
#include "../../plugins/components/Submodel.h"
#include "../../plugins/components/Match.h"
#include "../../plugins/components/PickUp.h"
#include "../../plugins/components/Remove.h"
#include "../../plugins/components/Search.h"
#include "../../plugins/components/Signal.h"
#include "../../plugins/components/Store.h"
#include "../../plugins/components/PickStation.h"
#include "../../plugins/components/Start.h"
#include "../../plugins/components/Stop.h"
#include "../../plugins/components/Unstore.h"
#include "../../plugins/components/Access.h"
#include "../../plugins/components/Exit.h"
#include "../../plugins/components/LSODE.h"
#include "../../plugins/components/MarkovChain.h"
#include "../../plugins/elements/Failure.h"
#include "../../plugins/elements/File.h"
#include "../../plugins/elements/Storage.h"
#include "../../plugins/elements/Sequence.h"
#include "../../plugins/elements/Schedule.h"
#include "../../plugins/elements/Variable.h"
#include "../../plugins/elements/EntityGroup.h"
#include "../../plugins/elements/Station.h"
#include "../../plugins/elements/Set.h"
#include "../util/Util.h"
```

8.280 PluginConnectorDummyImpl1.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
```

```
 * File:   PluginConnectorDummyImpl1.cpp
 * Author: rlcancian
 *
 * Created on 9 de Setembro de 2019, 19:24
 */

#include "PluginConnectorDummyImpl1.h"

// Model Components

#include "../../plugins/components/Create.h"
#include "../../plugins/components/Delay.h"
#include "../../plugins/components/Dispose.h"
#include "../../plugins/components/Seize.h"
#include "../../plugins/components/Release.h"
#include "../../plugins/components/Process.h"
#include "../../plugins/components/Assign.h"
#include "../../plugins/components/Record.h"
#include "../../plugins/components/Decide.h"
#include "../../plugins/components/Dummy.h"
#include "../../plugins/components/Route.h"
#include "../../plugins/components/Enter.h"
#include "../../plugins/components/Leave.h"
#include "../../plugins/components/Write.h"
#include "../../plugins/components/Hold.h"
#include "../../plugins/components/DropOff.h"
#include "../../plugins/components/Batch.h"
#include "../../plugins/components/Separate.h"
#include "../../plugins/components/Submodel.h"
#include "../../plugins/components/Match.h"
#include "../../plugins/components/PickUp.h"
#include "../../plugins/components/Remove.h"
//#include "../../plugins/components/Read.h"
#include "../../plugins/components/Search.h"
#include "../../plugins/components/Signal.h"
#include "../../plugins/components/Store.h"
#include "../../plugins/components/PickStation.h"
#include "../../plugins/components/Start.h"
#include "../../plugins/components/Stop.h"
#include "../../plugins/components/Unstore.h"
#include "../../plugins/components/Access.h"
#include "../../plugins/components/Exit.h"
#include "../../plugins/components/LSODE.h"
#include "../../plugins/components/MarkovChain.h"


// Model elements
#include "../../plugins/elements/Failure.h"
#include "../../plugins/elements/File.h"
#include "../../plugins/elements/Storage.h"
#include "../../plugins/elements/Sequence.h"
#include "../../plugins/elements/Schedule.h"
#include "../../plugins/elements/Variable.h"
#include "../../plugins/elements/EntityGroup.h"
#include "../../plugins/elements/Station.h"
#include "../../plugins/elements/Set.h"
#include "../../plugins/elements/Storage.h"
#include "../../plugins/elements/Sequence.h"
//#include "../../plugins/elements/Expression.h"
//#include "../../plugins/elements/Conveyor.h"
//#include "../../plugins/elements/Segment.h"

#include "../util/Util.h"

//namespace GenesysKernel {

PluginConnectorDummyImpl1::PluginConnectorDummyImpl1() {
}

Plugin* PluginConnectorDummyImpl1::check(const std::string dynamicLibraryFilename) {
    // \todo: not implemented yet!
    //dynamicLibraryFilename = ""; // just to use it
    return nullptr;
}

Plugin* PluginConnectorDummyImpl1::connect(const std::string dynamicLibraryFilename) {
    std::string fn = getFileName(dynamicLibraryFilename);
    StaticGetPluginInformation GetInfo = nullptr;
    Plugin* pluginResult = nullptr;
    // ODO: Remove this mostruosity and and trully dynamically loaded libraries

    // model elements


    //if (fn == "attribute.so")
    //    GetInfo = &Attribute::GetPluginInformation;
    if (fn == "assign.so")
```

```
        GetInfo = &Assign::GetPluginInformation;
    else if (fn == "entitygroup.so")
        GetInfo = &EntityGroup::GetPluginInformation;
        //  else if (fn == "ode.so")
        //      GetInfo = &OLD_ODEelement::GetPluginInformation;
    else if (fn == "queue.so")
        GetInfo = &Queue::GetPluginInformation;
    else if (fn == "resource.so")
        GetInfo = &Resource::GetPluginInformation;
    else if (fn == "set.so")
        GetInfo = &Set::GetPluginInformation;
    else if (fn == "station.so")
        GetInfo = &Station::GetPluginInformation;
    else if (fn == "variable.so")
        GetInfo = &Variable::GetPluginInformation;
        // model components
    else if (fn == "create.so")
        GetInfo = &Create::GetPluginInformation;
    else if (fn == "write.so")
        GetInfo = &Write::GetPluginInformation;
    else if (fn == "decide.so")
        GetInfo = &Decide::GetPluginInformation;
    else if (fn == "delay.so")
        GetInfo = &Delay::GetPluginInformation;
    else if (fn == "dispose.so")
        GetInfo = &Dispose::GetPluginInformation;
    else if (fn == "dummy.so")
        GetInfo = &Dummy::GetPluginInformation;
    else if (fn == "record.so")
        GetInfo = &Record::GetPluginInformation;
    else if (fn == "release.so")
        GetInfo = &Release::GetPluginInformation;
    else if (fn == "seize.so")
        GetInfo = &Seize::GetPluginInformation;
    else if (fn == "route.so")
        GetInfo = &Route::GetPluginInformation;
    else if (fn == "process.so")
        GetInfo = &Process::GetPluginInformation;
    else if (fn == "enter.so")
        GetInfo = &Enter::GetPluginInformation;
    else if (fn == "leave.so")
        GetInfo = &Leave::GetPluginInformation;
    else if (fn == "lsode.so")
        GetInfo = &LSODE::GetPluginInformation;
    else if (fn == "markovchain.so")
        GetInfo = &MarkovChain::GetPluginInformation;
    else if (fn == "hold.so")
        GetInfo = &Hold::GetPluginInformation;
    else if (fn == "schedule.so")
        GetInfo = &Schedule::GetPluginInformation;
    else if (fn == "dropoff.so")
        GetInfo = &DropOff::GetPluginInformation;
    else if (fn == "batch.so")
        GetInfo = &Batch::GetPluginInformation;
    else if (fn == "separate.so")
        GetInfo = &Separate::GetPluginInformation;
        //else if (fn == "submodel.so")
        //    GetInfo = &Submodel::GetPluginInformation;
    else if (fn == "match.so")
        GetInfo = &Match::GetPluginInformation;
    else if (fn == "pickup.so")
        GetInfo = &PickUp::GetPluginInformation;
        //else if (fn == "read.so")
        //    GetInfo = &Read::GetPluginInformation;
    else if (fn == "remove.so")
        GetInfo = &Remove::GetPluginInformation;
    else if (fn == "search.so")
        GetInfo = &Search::GetPluginInformation;
    else if (fn == "signal.so")
        GetInfo = &Signal::GetPluginInformation;
    else if (fn == "store.so")
        GetInfo = &Store::GetPluginInformation;
    else if (fn == "unstore.so")
        GetInfo = &Unstore::GetPluginInformation;
        //else if (fn == "expression.so")
        //  GetInfo = &Expression::GetPluginInformation;
    else if (fn == "failure.so")
        GetInfo = &Failure::GetPluginInformation;
    else if (fn == "file.so")
        GetInfo = &File::GetPluginInformation;
    else if (fn == "storage.so")
        GetInfo = &Storage::GetPluginInformation;
    else if (fn == "pickstation.so")
        GetInfo = &PickStation::GetPluginInformation;
    else if (fn == "sequence.so")
        GetInfo = &Sequence::GetPluginInformation;
    else if (fn == "access.so")
```

```
        GetInfo = &Access::GetPluginInformation;
    else if (fn == "exit.so")
        GetInfo = &Exit::GetPluginInformation;
    else if (fn == "start.so")
        GetInfo = &Start::GetPluginInformation;
    else if (fn == "stop.so")
        GetInfo = &Stop::GetPluginInformation;
    //else if (fn == "conveyour.so")
    //    GetInfo = &Conveyour::GetPluginInformation;
    //else if (fn == "segment.so")
    //    GetInfo = &Segment::GetPluginInformation;

    //else if (fn==="")

    if (GetInfo != nullptr) {
        pluginResult = new Plugin(GetInfo);
    }
    return pluginResult;
}

bool PluginConnectorDummyImpl1::disconnect(const std::string dynamicLibraryFilename) {
    // \TODO: To implement!
    //dynamicLibraryFilename = ""; // just to use ut
    return true;
}

bool PluginConnectorDummyImpl1::disconnect(Plugin* plugin) {
    // \TODO: To implement!
    return true;
}

//namespace\\}
```

8.281 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginConnector↩ DummyImpl1.h File Reference

`#include "PluginConnector_if.h"`

Include dependency graph for PluginConnectorDummyImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PluginConnectorDummyImpl1

8.282 PluginConnectorDummyImpl1.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   PluginConnectorDummyImpl1.h
00009  * Author: rlcancian
00010  *
00011  * Created on 9 de Setembro de 2019, 19:24
00012  */
00013
00014 #ifndef PLUGINCONNECTORDUMMYIMPL1_H
00015 #define PLUGINCONNECTORDUMMYIMPL1_H
00016
00017 #include "PluginConnector_if.h"
00018 //namespace GenesysKernel {
00019
00020 class PluginConnectorDummyImpl1 : public PluginConnector_if {
00021 public:
00022     PluginConnectorDummyImpl1();
00023     virtual ~PluginConnectorDummyImpl1() = default;
00024 public:
00025     virtual Plugin* check(const std::string dynamicLibraryFilename);
00026     virtual Plugin* connect(const std::string dynamicLibraryFilename);
00027     virtual bool disconnect(const std::string dynamicLibraryFilename);
00028     virtual bool disconnect(Plugin* plugin);
00029 private:
00030
00031 };
00032 //namespace\\}
00033 #endif /* PLUGINCONNECTORDUMMYIMPL1_H */
00034
```

8.283 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginInformation.cpp File Reference

`#include "PluginInformation.h"`

Include dependency graph for PluginInformation.cpp:

8.284 PluginInformation.cpp

```
00001 /*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PluginInformation.cpp
 * Author: rlcancian
 *
 * Created on 9 de Setembro de 2019, 20:02
 */

#include "PluginInformation.h"

//using namespace GenesysKernel;

PluginInformation::PluginInformation(std::string pluginTypename, StaticLoaderComponentInstance componentloader) {
    this->_componentloader = componentloader;
    this->_elementloader = nullptr;
    this->_isComponent = true;
    this->_pluginTypename = pluginTypename;
}

PluginInformation::PluginInformation(std::string pluginTypename, StaticLoaderElementInstance elementloader) {
    this->_componentloader = nullptr;
    this->_elementloader = elementloader;
    this->_isComponent = false;
    this->_pluginTypename = pluginTypename;
}

StaticLoaderElementInstance PluginInformation::getElementLoader() const {
    return _elementloader;
}

StaticLoaderComponentInstance PluginInformation::GetComponentLoader() const {
    return _componentloader;
}

bool PluginInformation::isGenerateReport() const {
    return _generateReport;
}

bool PluginInformation::isComponent() const {
    return _isComponent;
}

bool PluginInformation::isSendTransfer() const {
    return _sendTransfer;
}

bool PluginInformation::isReceiveTransfer() const {
    return _receiveTransfer;
}

bool PluginInformation::isSink() const {
    return _isSink;
}

bool PluginInformation::isSource() const {
    return _isSource;
}

std::string PluginInformation::getObservation() const {
    return _observation;
}

std::string PluginInformation::getVersion() const {
    return _version;
}

std::string PluginInformation::getDate() const {
    return _date;
}

std::string PluginInformation::getAuthor() const {
    return _author;
}

std::string PluginInformation::getPluginTypename() const {
    return _pluginTypename;
}

void PluginInformation::insertDynamicLibFileDependence(std::string filename) {
    _dynamicLibFilenameDependencies->insert(_dynamicLibFilenameDependencies->end(), filename);
}
```

```cpp

void PluginInformation::setDynamicLibFilenameDependencies(std::list<std::string>* dynamicLibFilenameDependencies) {
    this->_dynamicLibFilenameDependencies = dynamicLibFilenameDependencies;
}

std::list<std::string>* PluginInformation::getDynamicLibFilenameDependencies() const {
    return _dynamicLibFilenameDependencies;
}

void PluginInformation::setGenerateReport(bool generateReport) {
    this->_generateReport = generateReport;
}

void PluginInformation::setSendTransfer(bool sendTransfer) {
    this->_sendTransfer = sendTransfer;
}

void PluginInformation::setReceiveTransfer(bool receiveTransfer) {
    this->_receiveTransfer = receiveTransfer;
}

void PluginInformation::setSink(bool Sink) {
    _isSink = Sink;
}

void PluginInformation::setSource(bool Source) {
    _isSource = Source;
}

void PluginInformation::setObservation(std::string observation) {
    this->_observation = observation;
}

void PluginInformation::setVersion(std::string version) {
    this->_version = version;
}

void PluginInformation::setDate(std::string date) {
    this->_date = date;
}

void PluginInformation::setAuthor(std::string author) {
    this->_author = author;
}

void PluginInformation::setMaximumOutputs(unsigned short _maximumOutputs) {
    this->_maximumOutputs = _maximumOutputs;
}

unsigned short PluginInformation::getMaximumOutputs() const {
    return _maximumOutputs;
}

void PluginInformation::setMinimumOutputs(unsigned short _minimumOutputs) {
    this->_minimumOutputs = _minimumOutputs;
}

unsigned short PluginInformation::getMinimumOutputs() const {
    return _minimumOutputs;
}

void PluginInformation::setMaximumInputs(unsigned short _maximumInputs) {
    this->_maximumInputs = _maximumInputs;
}

unsigned short PluginInformation::getMaximumInputs() const {
    return _maximumInputs;
}

void PluginInformation::setMinimumInputs(unsigned short _minimumInputs) {
    this->_minimumInputs = _minimumInputs;
}

unsigned short PluginInformation::getMinimumInputs() const {
    return _minimumInputs;
}

void PluginInformation::setDescriptionHelp(std::string _descriptionHelp) {
    this->_descriptionHelp = _descriptionHelp;
}

std::string PluginInformation::getDescriptionHelp() const {
    return _descriptionHelp;
}

void PluginInformation::setFields(std::map<std::string, std::string>* _fields) {
```

```
    this->_fields = _fields;
}

std::map<std::string, std::string>* PluginInformation::getFields() const {
    return _fields;
}
```

8.285 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginInformation.h File Reference

```
#include <map>
#include <list>
#include <string>
```

Include dependency graph for PluginInformation.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PluginInformation

Typedefs

- typedef ModelComponent *(* StaticLoaderComponentInstance) (Model *, std::map< std::string, std::string > *)
- typedef ModelElement *(* StaticLoaderElementInstance) (Model *, std::map< std::string, std::string > *)
- typedef PluginInformation *(* StaticGetPluginInformation) ()

8.285.1 Typedef Documentation

8.285.1.1 StaticGetPluginInformation `typedef PluginInformation*(* StaticGetPluginInformation) ()`

Definition at line 31 of file PluginInformation.h.

8.285.1.2 StaticLoaderComponentInstance `typedef ModelComponent*(* StaticLoaderComponent↩ Instance) (Model *, std::map< std::string, std::string > *)`

Definition at line 28 of file PluginInformation.h.

8.285.1.3 StaticLoaderElementInstance `typedef ModelElement*(* StaticLoaderElementInstance) (Model *, std::map< std::string, std::string > *)`

Definition at line 29 of file PluginInformation.h.

8.286 PluginInformation.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   PluginInformation.h
00009  * Author: rlcancian
00010  *
00011  * Created on 9 de Setembro de 2019, 20:02
00012  */
00013
00014 #ifndef PLUGININFORMATION_H
00015 #define PLUGININFORMATION_H
00016
00017 #include <map>
00018 #include <list>
00019 #include <string>
00020
00021 //namespace GenesysKernel {
00022 class ModelElement;
00023 class ModelComponent;
00024 class Model;
00025 class ElementManager;
00026
00027 // \todo: the following 2 types were diffferent but now on they are the same and should be merged
00028 typedef ModelComponent* (*StaticLoaderComponentInstance)(Model*, std::map<std::string, std::string>*);
00029 typedef ModelElement* (*StaticLoaderElementInstance)(Model*, std::map<std::string, std::string>*);
00030 class PluginInformation;
00031 typedef PluginInformation* (*StaticGetPluginInformation)();
00032
00033 class PluginInformation {
00034 public:
00035     PluginInformation(std::string pluginTypename, StaticLoaderComponentInstance componentloader);
00036     PluginInformation(std::string pluginTypename, StaticLoaderElementInstance elementloader);
00037 public:
00038     // gets
00039     StaticLoaderElementInstance getElementLoader() const;
00040     StaticLoaderComponentInstance GetComponentLoader() const;
00041     bool isGenerateReport() const;
00042     bool isComponent() const;
00043     bool isSendTransfer() const;
00044     bool isReceiveTransfer() const;
```

```
    bool isSink() const;
    bool isSource() const;
    std::string getObservation() const;
    std::string getVersion() const;
    std::string getDate() const;
    std::string getAuthor() const;
    std::string getPluginTypename() const;
    // sets
    void insertDynamicLibFileDependence(std::string filename);
    void setDynamicLibFilenameDependencies(std::list<std::string>* dynamicLibFilenameDependencies);
    std::list<std::string>* getDynamicLibFilenameDependencies() const;
    void setGenerateReport(bool generateReport);
    void setSendTransfer(bool sendTransfer);
    void setReceiveTransfer(bool receiveTransfer);
    void setSink(bool Sink);
    void setSource(bool Source);
    void setObservation(std::string observation);
    void setVersion(std::string version);
    void setDate(std::string date);
    void setAuthor(std::string author);
    void setMaximumOutputs(unsigned short _maximumOutputs);
    unsigned short getMaximumOutputs() const;
    void setMinimumOutputs(unsigned short _minimumOutputs);
    unsigned short getMinimumOutputs() const;
    void setMaximumInputs(unsigned short _maximumInputs);
    unsigned short getMaximumInputs() const;
    void setMinimumInputs(unsigned short _minimumInputs);
    unsigned short getMinimumInputs() const;
    void setDescriptionHelp(std::string _descriptionHelp);
    std::string getDescriptionHelp() const;
    void setFields(std::map<std::string, std::string>* _fiewlds);
    std::map<std::string, std::string>* getFields() const;
public:
private:
    std::string _author = "prof. Dr. Ing. Rafael Luiz Cancian";
    std::string _date = "01/08/2018";
    std::string _version = "0.9.1";
    std::string _observation = "First implementation not fully completed nor tested. Use with caution.";
    std::string _descriptionHelp = "";
    bool _isSource = false;
    bool _isSink = false;
    bool _receiveTransfer = false;
    bool _sendTransfer = false;
    bool _generateReport = false;
    std::list<std::string>* _dynamicLibFilenameDependencies = new std::list<std::string>();
    std::map<std::string, std::string>* _fields = new std::map<std::string, std::string>();
    // set from constructor
    std::string _pluginTypename;
    bool _isComponent = false;
    unsigned short _minimumInputs = 1;
    unsigned short _maximumInputs = 1;
    unsigned short _minimumOutputs = 1;
    unsigned short _maximumOutputs = 1;
    StaticLoaderComponentInstance _componentloader;
    StaticLoaderElementInstance _elementloader;
};
//namespace\\}

#endif /* PLUGININFORMATION_H */
```

8.287 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginManager.cpp File Reference

```
#include "PluginManager.h"
#include "Simulator.h"
#include "../TraitsKernel.h"
#include "Attribute.h"
#include "Counter.h"
#include "EntityType.h"
#include "StatisticsCollector.h"
```

Include dependency graph for PluginManager.cpp:

8.288 PluginManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PluginManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 30 de Maio de 2019, 17:49
 */

#include "PluginManager.h"
#include "Simulator.h"
#include "../TraitsKernel.h"
#include "Attribute.h"
#include "Counter.h"
#include "EntityType.h"
#include "StatisticsCollector.h"

//using namespace GenesysKernel;

PluginManager::PluginManager(Simulator* simulator) {
    _simulator = simulator;
    this->_pluginConnector = new TraitsKernel<PluginConnector_if>::Implementation();
    this->_insertDefaultKernelElements();
}

void PluginManager::_insertDefaultKernelElements() {
    StaticGetPluginInformation GetInfo;
    GetInfo = &EntityType::GetPluginInformation;
    _plugins->insert(new Plugin(GetInfo));
    GetInfo = &Attribute::GetPluginInformation;
    _plugins->insert(new Plugin(GetInfo));
    GetInfo = &Counter::GetPluginInformation;
    _plugins->insert(new Plugin(GetInfo));
    GetInfo = &StatisticsCollector::GetPluginInformation;
    _plugins->insert(new Plugin(GetInfo));
}

//bool PluginManager::check(Plugin* plugin){
//    return true;
//}

bool PluginManager::_insert(Plugin* plugin) {
    PluginInformation *plugInfo = plugin->getPluginInfo();
    if (plugin->isIsValidPlugin() && plugInfo != nullptr) {
        std::string msg = "Inserting ";
        if (plugInfo->isComponent())
            msg += "component";
        else
            msg += "element";
        msg += " plugin \"" + plugin->getPluginInfo()->getPluginTypename() + "\"";
        _simulator->getTracer()->trace(Util::TraceLevel::L8_detailed, msg);
        // insert all dependencies before to insert this plugin
        bool allDependenciesInserted = true;
        if (plugInfo->getDynamicLibFilenameDependencies()->size() > 0) {
            Util::IncIndent();
            {
                _simulator->getTracer()->trace(Util::TraceLevel::L8_detailed, "Inserting dependencies...");
                Util::IncIndent();
                {
                    for (std::string str : *plugInfo->getDynamicLibFilenameDependencies()) {
                        allDependenciesInserted &= (this->insert(str) != nullptr);
```

```
                    }
                }
                Util::DecIndent();
            }
            Util::DecIndent();
        }
        if (!allDependenciesInserted) {
            _simulator->getTracer()->trace(Util::TraceLevel::L3_errorRecover, "Plugin dependencies could not be inserted; therefore, plugin will not be inserted");
            return false;
        }
        if (this->find(plugInfo->getPluginTypename()) != nullptr) { // plugin alread exists
            Util::IncIndent();
            _simulator->getTracer()->trace(Util::TraceLevel::L8_detailed, "Plugin alread exists and was not inserted again");
            Util::DecIndent();
            return false;
        }
        _plugins->insert(plugin);
        Util::IncIndent();
        this->_simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Plugin successfully inserted");
        Util::DecIndent();
        return true;
    } else {
        Util::IncIndent();
        this->_simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Plugin could not be inserted");
        plugin->~Plugin(); // destroy the invalid plugin
        Util::DecIndent();
        return false;
    }
}

bool PluginManager::check(std::string dynamicLibraryFilename) {
    Plugin* plugin;
    try {
        plugin = _pluginConnector->check(dynamicLibraryFilename);
    } catch (...) {
        return false;
    }
    return (plugin != nullptr);
}

Plugin* PluginManager::insert(std::string dynamicLibraryFilename) {
    Plugin* plugin;
    try {
        plugin = _pluginConnector->connect(dynamicLibraryFilename);
        if (plugin != nullptr)
            _insert(plugin);
        else {
            _simulator->getTracer()->trace(Util::TraceLevel::L3_errorRecover, "Plugin from file \"" + dynamicLibraryFilename + "\" could not be loaded.");
        }
    } catch (...) {

        return nullptr;
    }
    return plugin;
}

bool PluginManager::remove(std::string dynamicLibraryFilename) {

    Plugin* pi = this->find(dynamicLibraryFilename);
    return remove(pi);
}

bool PluginManager::remove(Plugin* plugin) {
    if (plugin != nullptr) {
        _plugins->remove(plugin);
        try {
            _pluginConnector->disconnect(plugin);
        } catch (...) {
            return false;
        }
        _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Plugin successfully removed");
        return true;
    }
    _simulator->getTracer()->trace(Util::TraceLevel::L2_results, "Plugin could not be removed");
    return false;
}

Plugin* PluginManager::find(std::string pluginTypeName) {
    for (std::list<Plugin*>::iterator it = this->_plugins->list()->begin(); it != _plugins->list()->end(); it++) {
        if ((*it)->getPluginInfo()->getPluginTypename() == pluginTypeName) {

```

```
            return (*it);
        }
    }
    return nullptr;
}

Plugin* PluginManager::front() {

    return this->_plugins->front();
}

Plugin* PluginManager::next() {

    return _plugins->next();
}

Plugin* PluginManager::last() {
    return this->_plugins->last();
}
```

8.289 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/PluginManager.h File Reference

```
#include "../util/List.h"
#include "Plugin.h"
#include "PluginConnector_if.h"
```

Include dependency graph for PluginManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PluginManager

8.290 PluginManager.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PluginManager.h
 * Author: rafael.luiz.cancian
 *
 * Created on 30 de Maio de 2019, 17:49
 */

#ifndef PLUGINMANAGER_H
#define PLUGINMANAGER_H

#include "../util/List.h"
#include "Plugin.h"
#include "PluginConnector_if.h"

//namespace GenesysKernel {
class Simulator;

class PluginManager {
public:
    PluginManager(Simulator* simulator);
    virtual ~PluginManager() = default;
public:
    bool check(const std::string dynamicLibraryFilename);
    Plugin* insert(const std::string dynamicLibraryFilename);
    bool remove(const std::string dynamicLibraryFilename);
    bool remove(Plugin* plugin);
    Plugin* find(std::string pluginTypeName);
public:
    Plugin* front();
    Plugin* next();
    Plugin* last();
private:
    bool _insert(Plugin* plugin);
    void _insertDefaultKernelElements();
private:
    List<Plugin*>* _plugins = new List<Plugin*>();
    Simulator* _simulator;
    PluginConnector_if* _pluginConnector;
};
//namespace\\}
#endif /* PLUGINMANAGER_H */

```

8.291 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/ScenarioExperiment↩ _if.h File Reference

Classes

- class ScenarioExperiment_if

8.292 ScenarioExperiment_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ScenarioExperiment_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 10 de Outubro de 2018, 14:52
 */

#ifndef SCENARIOEXPERIMENT_IF_H
#define SCENARIOEXPERIMENT_IF_H

```

```
00017 class ScenarioExperiment_if {
00018 };
00019
00020 #endif /* SCENARIOEXPERIMENT_IF_H */
00021
```

8.293 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationControl.cpp File Reference

`#include "SimulationControl.h"`

Include dependency graph for SimulationControl.cpp:

8.294 SimulationControl.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   SimulationControl.cpp
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 10 de Outubro de 2018, 18:01
00012  */
00013
00014 #include "SimulationControl.h"
00015
00016 //using namespace GenesysKernel;
00017
00018 SimulationControl::SimulationControl(std::string type, std::string name, GetterMember getterMember,
       SetterMember setterMember, std::string parent) : SimulationResponse(type, name, getterMember, parent)
       {
00019     this->_setMemberFunction = setterMember;
00020 }
00021
00022 std::string SimulationControl::show() {
00023     return "name=" + this->_name + ", type=" + this->_type;
00024 }
00025
00026 void SimulationControl::setValue(double value) {
00027     this->_setMemberFunction(value);
00028 }
00029
```

8.295 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationControl.h File Reference

`#include "SimulationResponse.h"`

`#include "DefineGetterSetter.h"`
Include dependency graph for SimulationControl.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationControl

8.296 SimulationControl.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   SimulationControl.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 10 de Outubro de 2018, 18:01
00012  */
00013
00014 #ifndef SIMULATIONCONTROL_H
00015 #define SIMULATIONCONTROL_H
00016
00017 #include "SimulationResponse.h"
00018 #include "DefineGetterSetter.h"
00019 //namespace GenesysKernel {
00020
00024 class SimulationControl : public SimulationResponse {
00025 public:
00026     SimulationControl(std::string type, std::string name, GetterMember getterMember, SetterMember
       setterMember, std::string parent = "");
00027     virtual ~SimulationControl() = default;
00028 public:
00029     std::string show();
00030 public:
00031     void setValue(double value);
00032 private:
00033     SetterMember _setMemberFunction;
00034     //SetterMemberString  _setMemberStringFunction;
00035 };
00036 //namespace\\}
00037 #endif /* SIMULATIONCONTROL_H */
00038
```

8.297 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationExperiment.cpp File Reference

```
#include "SimulationExperiment.h"
```

Include dependency graph for SimulationExperiment.cpp:

8.298 SimulationExperiment.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationExperiment.cpp
 * Author: rlcancian
 *
 * Created on 4 de maio de 2021, 15:10
 */

#include "SimulationExperiment.h"

SimulationExperiment::SimulationExperiment() {
}


```

8.299 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationExperiment.h File Reference

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationExperiment

8.300 SimulationExperiment.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationExperiment.h
 * Author: rlcancian
 *
 * Created on 4 de maio de 2021, 15:10
 */

#ifndef SIMULATIONEXPERIMENT_H
#define SIMULATIONEXPERIMENT_H

class SimulationExperiment {
public:
    SimulationExperiment();
    virtual ~SimulationExperiment() = default;
private:

};

#endif /* SIMULATIONEXPERIMENT_H */

```

8.301 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationReporter_↩ if.h File Reference

`#include "../util/List.h"`

Include dependency graph for SimulationReporter_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationReporter_if

8.302 SimulationReporter_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationReporter_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 8 de Agosto de 2018, 10:56
 */

#ifndef SIMULATIONREPORTER_IF_H
#define SIMULATIONREPORTER_IF_H

#include "../util/List.h"
//#include "StatisticsCollector.h"

class SimulationReporter_if {
public:
    virtual void showReplicationStatistics() = 0;
    virtual void showSimulationStatistics() = 0;
    virtual void showSimulationResponses() = 0;
    virtual void showSimulationControls() = 0;
};

#endif /* SIMULATIONREPORTER_IF_H */

```

8.303 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationReporter↩ DefaultImpl1.cpp File Reference

```
#include "SimulationReporterDefaultImpl1.h"
#include <assert.h>
#include <iomanip>
#include <iostream>
#include "Counter.h"
```

Include dependency graph for SimulationReporterDefaultImpl1.cpp:

8.304 SimulationReporterDefaultImpl1.cpp

```cpp
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationReporterDefaultImpl1.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 8 de Agosto de 2018, 10:59
 */

#include "SimulationReporterDefaultImpl1.h"
#include <assert.h>
#include <iomanip>
#include <iostream>
#include "Counter.h"

//using namespace GenesysKernel;

SimulationReporterDefaultImpl1::SimulationReporterDefaultImpl1(ModelSimulation* simulation, Model* model, List<ModelElement*>* statsCountersSimulation) {
    _simulation = simulation;
    _model = model;
    _statsCountersSimulation = statsCountersSimulation;
}

void SimulationReporterDefaultImpl1::showSimulationControls() {
    _model->getTracer()->traceReport("Simulation Controls:");
    Util::IncIndent();
    {
        for (std::list<SimulationControl*>::iterator it = _model->getControls()->list()->begin(); it != _model->getControls()->list()->end(); it++) {
            SimulationControl* control = (*it);
            _model->getTracer()->traceReport(control->getName() + ": " + std::to_string(control->getValue()));
        }
    }
    Util::DecIndent();
}

void SimulationReporterDefaultImpl1::showReplicationStatistics() {
    _model->getTracer()->traceReport("");
    _model->getTracer()->traceReport("Begin of Report for replication " + std::to_string(_simulation->getCurrentReplicationNumber()) + " of " + std::to_string(_model->getSimulation()->getNumberOfReplications()));
    /* \todo: StatisticsCollector and Counter should NOT be special classes. It should iterate classes looking for classes that can generate reports.
     StatisticsCollector and Counter should ovveride an inherited attribute from ModelElement to specify they generate report information
     look for _generateReportInformation = true;  using bool generateReportInformation() const;
     */
    const std::string UtilTypeOfStatisticsCollector = Util::TypeOf<StatisticsCollector>();
    const std::string UtilTypeOfCounter = Util::TypeOf<Counter>();
    // runs over all elements and list the statistics for each one, and then the statistics with no parent
    Util::IncIndent();
    if (_simulation->isShowSimulationControlsInReport())
        this->showSimulationControls();
    // copy the ist of statistics and counters into a single new list
    std::list<ModelElement*>* statisticsAndCounters = new std::list<ModelElement*>(*(_model->getElements()->getElementList(UtilTypeOfStatisticsCollector)->list()));
    std::list<ModelElement*>* counters = new std::list<ModelElement*>(*(_model->getElements()->getElementList(UtilTypeOfCounter)->list()));
    statisticsAndCounters->merge(*counters);
    // organizes statistics into a map of maps
    std::map< std::string, std::map<std::string, std::list<ModelElement*>*>* >* mapMapTypeStat = new std::map<std::string, std::map<std::string, std::list<ModelElement*>*>*>();

    for (std::list<ModelElement*>::iterator it = statisticsAndCounters->begin(); it != statisticsAndCounters->end(); it++) {
        std::string parentName, parentTypename;
        ModelElement* statOrCnt = (*it);
        //std::cout « statOrCnt->getName() « ": " « statOrCnt->getTypename() « std::endl;
        if ((*it)->getClassname() == UtilTypeOfStatisticsCollector) {
            StatisticsCollector* stat = dynamic_cast<StatisticsCollector*> (statOrCnt);
            parentName = stat->getParent()->getName();
            parentTypename = stat->getParent()->getClassname();
        } else {
            if ((*it)->getClassname() == UtilTypeOfCounter) {
                Counter* cnt = dynamic_cast<Counter*> (statOrCnt);
                parentName = cnt->getParent()->getName();
                parentTypename = cnt->getParent()->getClassname();

```

```
            }
        }
        // look for key=parentTypename
        std::map<std::string, std::map<std::string, std::list<ModelElement*>*>*>::iterator mapMapIt = mapMapTypeStat->find(parentTypename);
        if (mapMapIt == mapMapTypeStat->end()) { // parentTypename does not exists in map. Include it.
            std::pair< std::string, std::map<std::string, std::list<ModelElement*>*>* >* newPair = new std::pair<std::string, std::map<std::string, std::list<ModelElement*>*>*>(parentTypename, new std::map<std::string, std::list<ModelElement*>*>());
            mapMapTypeStat->insert(*newPair);
            mapMapIt = mapMapTypeStat->find(parentTypename); // find again. Now it will.
        }
        //assert(mapMapIt != mapMapTypeStat->end());
        std::map<std::string, std::list<ModelElement*>*>* mapTypeStat = (*mapMapIt).second;
        assert(mapTypeStat != nullptr);
        // look for key=parentName
        std::map<std::string, std::list<ModelElement*>*>::iterator mapIt = mapTypeStat->find(parentName);
        if (mapIt == mapTypeStat->end()) { // parentTypename does not exists in map. Include it.
            std::pair< std::string, std::list<ModelElement*>* >* newPair = new std::pair<std::string, std::list<ModelElement*>*>(parentName, new std::list<ModelElement*>());
            mapTypeStat->insert(*newPair);
            mapIt = mapTypeStat->find(parentName); // find again. Now it will.
        }
        // get the list and insert the stat in that list
        std::list<ModelElement*>* listStatAndCount = (*mapIt).second;
        assert(listStatAndCount != nullptr);
        listStatAndCount->insert(listStatAndCount->end(), statOrCnt);
        //_model->getTraceManager()->traceReport(parentTypename + " -> " + parentName + " -> " + stat->show());
    }
    // now runs over that map of maps showing the statistics
    Util::IncIndent();
    Util::IncIndent();
    _model->getTracer()->traceReport(Util::SetW("name", _nameW) + Util::SetW("elems", _w) + Util::SetW("min", _w) + Util::SetW("max", _w) + Util::SetW("average", _w) + Util::SetW("variance", _w) + Util::SetW("stddev", _w) + Util::SetW("varCoef", _w) + Util::SetW("confInterv", _w) + Util::SetW("confLevel", _w));
    Util::DecIndent();
    Util::DecIndent();
    for (auto const mapmapItem : *mapMapTypeStat) {
        _model->getTracer()->traceReport("Statistics for " + mapmapItem.first + ":");
        Util::IncIndent();
        {
            for (auto const mapItem : *(mapmapItem.second)) {
                _model->getTracer()->traceReport(mapItem.first + ":");
                Util::IncIndent();
                {
                    //                  _model->getTracer()->traceReport(Util::SetW("name", _nameW) + Util::SetW("elems", _w) + Util::SetW("min", _w) + Util::SetW("max", _w) + Util::SetW("average", _w) + Util::SetW("variance", _w) + Util::SetW("stddev", _w) + Util::SetW("varCoef", _w) + Util::SetW("confInterv", _w) + Util::SetW("confLevel", _w));
                    for (ModelElement * const item : *(mapItem.second)) {
                        if (item->getClassname() == UtilTypeOfStatisticsCollector) {
                            Statistics_if* stat = dynamic_cast<StatisticsCollector*> (item)->getStatistics();
                            _model->getTracer()->traceReport(Util::TraceLevel::L2_results,
                                    Util::SetW(item->getName() + std::string(_nameW, '.'), _nameW - 1) + " " +
                                    Util::SetW(std::to_string(stat->numElements()), _w) +
                                    Util::SetW(std::to_string(stat->min()), _w) +
                                    Util::SetW(std::to_string(stat->max()), _w) +
                                    Util::SetW(std::to_string(stat->average()), _w) +
                                    Util::SetW(std::to_string(stat->variance()), _w) +
                                    Util::SetW(std::to_string(stat->stddeviation()), _w) +
                                    Util::SetW(std::to_string(stat->variationCoef()), _w) +
                                    Util::SetW(std::to_string(stat->halfWidthConfidenceInterval()), _w) +
                                    Util::SetW(std::to_string(stat->getConfidenceLevel()), _w)
                                    );
                        } else {
                            if (item->getClassname() == UtilTypeOfCounter) {
                                Counter* count = dynamic_cast<Counter*> (item);
                                _model->getTracer()->traceReport(Util::TraceLevel::L2_results,
                                        Util::SetW(count->getName() + std::string(_nameW, '.'), _nameW - 1) + " " +
                                        Util::SetW(std::to_string(count->getCountValue()), _w)
                                        );
                            }
                        }
                    }
                }
                Util::DecIndent();
            }
        }
        Util::DecIndent();
    }
```

```
    if (_simulation->isShowSimulationResposesInReport())
        this->showSimulationResponses();
    Util::DecIndent();
    _model->getTracer()->traceReport("End of Report for replication " + std::to_string(_simulation->getCurrentReplicationNumber()) + " of " + std::to_string(_model->getSimulation()->getNumberOfReplications()));
    _model->getTracer()->traceReport("------------------------------------");
}

void SimulationReporterDefaultImpl1::showSimulationResponses() {
    _model->getTracer()->traceReport("Simulation Responses:");
    Util::IncIndent();
    {
        for (std::list<SimulationResponse*>::iterator it = _model->getResponses()->list()->begin(); it != _model->getResponses()->list()->end(); it++) {
            SimulationResponse* response = (*it);
            _model->getTracer()->traceReport(response->getName() + ": " + std::to_string(response->getValue()));
        }
    }
    Util::DecIndent();
}

void SimulationReporterDefaultImpl1::showSimulationStatistics() {//List<StatisticsCollector*>* cstatsSimulation) {
    _model->getTracer()->traceReport("");
    _model->getTracer()->traceReport("Begin of Report for Simulation (based on " + std::to_string(_model->getSimulation()->getNumberOfReplications()) + " replications)");
    const std::string UtilTypeOfStatisticsCollector = Util::TypeOf<StatisticsCollector>();
    const std::string UtilTypeOfCounter = Util::TypeOf<Counter>();
    // runs over all elements and list the statistics for each one, and then the statistics with no parent
    Util::IncIndent();
    // COPY the list of statistics and counters into a single new list
    //std::list<ModelElement*>* statisticsAndCounters = //new std::list<ModelElement*>(*(this->_statsCountersSimulation->list()));
    // organizes statistics into a map of maps
    std::map< std::string, std::map<std::string, std::list<ModelElement*>*>* >* mapMapTypeStat = new std::map<std::string, std::map<std::string, std::list<ModelElement*>*>*>();

    for (std::list<ModelElement*>::iterator it = _statsCountersSimulation->list()->begin(); it != _statsCountersSimulation->list()->end(); it++) {
        std::string parentName, parentTypename;
        ModelElement* statOrCnt = (*it);
        //std::cout « statOrCnt->getName() « ": " « statOrCnt->getTypename() « std::endl;
        if ((*it)->getClassname() == UtilTypeOfStatisticsCollector) {
            StatisticsCollector* stat = dynamic_cast<StatisticsCollector*> (statOrCnt);
            parentName = stat->getParent()->getName();
            parentTypename = stat->getParent()->getClassname();
        } else {
            if ((*it)->getClassname() == UtilTypeOfCounter) {
                Counter* cnt = dynamic_cast<Counter*> (statOrCnt);
                parentName = cnt->getParent()->getName();
                parentTypename = cnt->getParent()->getClassname();

            }
        }
        // look for key=parentTypename
        std::map<std::string, std::map<std::string, std::list<ModelElement*>*>*>::iterator mapMapIt = mapMapTypeStat->find(parentTypename);
        if (mapMapIt == mapMapTypeStat->end()) { // parentTypename does not exists in map. Include it.
            std::pair< std::string, std::map<std::string, std::list<ModelElement*>*>* >* newPair = new std::pair<std::string, std::map<std::string, std::list<ModelElement*>*>*>(parentTypename, new std::map<std::string, std::list<ModelElement*>*>());
            mapMapTypeStat->insert(*newPair);
            mapMapIt = mapMapTypeStat->find(parentTypename); // find again. Now it will.
        }
        assert(mapMapIt != mapMapTypeStat->end());
        std::map<std::string, std::list<ModelElement*>*>* mapTypeStat = (*mapMapIt).second;
        assert(mapTypeStat != nullptr);
        // look for key=parentName
        std::map<std::string, std::list<ModelElement*>*>::iterator mapIt = mapTypeStat->find(parentName);
        if (mapIt == mapTypeStat->end()) { // parentTypename does not exists in map. Include it.
            std::pair< std::string, std::list<ModelElement*>* >* newPair = new std::pair<std::string, std::list<ModelElement*>*>(parentName, new std::list<ModelElement*>());
            mapTypeStat->insert(*newPair);
            mapIt = mapTypeStat->find(parentName); // find again. Now it will.
        }
        // get the list and insert the stat in that list
        std::list<ModelElement*>* listStat = (*mapIt).second;
        listStat->insert(listStat->end(), statOrCnt);
        //_model->getTraceManager()->traceReport(parentTypename + " -> " + parentName + " -> " + stat->show());
    }
    // now runs over that map of maps showing the statistics
    //int w = 12;
```

```
    Util::IncIndent();
    Util::IncIndent();
    _model->getTracer()->traceReport(Util::SetW("name", _nameW) + Util::SetW("elems", _w) + Util::SetW("min", _w) + Util::SetW("max", _w) + Util::SetW("average", _w) + Util::SetW("variance", _w) + Util::SetW("stddev", _w) + Util::SetW("varCoef", _w) + Util::SetW("confInterv", _w) + Util::SetW("confLevel", _w));
    Util::DecIndent();
    Util::DecIndent();
    for (auto const mapmapItem : *mapMapTypeStat) {
        _model->getTracer()->traceReport("Statistics for " + mapmapItem.first + ":");
        Util::IncIndent();
        {
            for (auto const mapItem : *(mapmapItem.second)) {
                _model->getTracer()->traceReport(mapItem.first + ":");
                Util::IncIndent();
                {
                    for (ModelElement * const item : *(mapItem.second)) {
                        if (item->getClassname() == UtilTypeOfStatisticsCollector) {
                            Statistics_if* stat = dynamic_cast<StatisticsCollector*>(item)->getStatistics();
                            _model->getTracer()->traceReport(Util::TraceLevel::L2_results,
                                    Util::SetW(item->getName() + std::string(_nameW, '.'), _nameW - 1) + " " +
                                    Util::SetW(std::to_string(stat->numElements()), _w) +
                                    Util::SetW(std::to_string(stat->min()), _w) +
                                    Util::SetW(std::to_string(stat->max()), _w) +
                                    Util::SetW(std::to_string(stat->average()), _w) +
                                    Util::SetW(std::to_string(stat->variance()), _w) +
                                    Util::SetW(std::to_string(stat->stddeviation()), _w) +
                                    Util::SetW(std::to_string(stat->variationCoef()), _w) +
                                    Util::SetW(std::to_string(stat->halfWidthConfidenceInterval()), _w) +
                                    Util::SetW(std::to_string(stat->getConfidenceLevel()), _w)
                                    );
                        } else {
                            if (item->getClassname() == UtilTypeOfCounter) {
                                Counter* cnt = dynamic_cast<Counter*> (item);
                                _model->getTracer()->traceReport(Util::TraceLevel::L2_results,
                                        Util::SetW(cnt->getName() + std::string(_nameW, '.'), _nameW - 1) + " " +
                                        Util::SetW(std::to_string(cnt->getCountValue()), _w)
                                        );
                            }
                        }
                    }
                }
                Util::DecIndent();
            }
        }
        Util::DecIndent();
    }

    Util::DecIndent();
    _model->getTracer()->traceReport("End of Report for Simulation");
}
```

8.305 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationReporterDefaultImpl1.h File Reference

```
#include "SimulationReporter_if.h"
#include "ModelSimulation.h"
#include "Model.h"
```

Include dependency graph for SimulationReporterDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationReporterDefaultImpl1

8.306 SimulationReporterDefaultImpl1.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationReporterDefaultImpl1.h
 * Author: rafael.luiz.cancian
 *
 * Created on 8 de Agosto de 2018, 10:59
 */

#ifndef SIMULATIONREPORTERDEFAULTIMPL1_H
#define SIMULATIONREPORTERDEFAULTIMPL1_H

#include "SimulationReporter_if.h"
#include "ModelSimulation.h"
#include "Model.h"

//namespace GenesysKernel {

class SimulationReporterDefaultImpl1 : public SimulationReporter_if {
public:
    SimulationReporterDefaultImpl1(ModelSimulation* simulation, Model* model, List<ModelElement*>*
 statsCountersSimulation);
    virtual ~SimulationReporterDefaultImpl1() = default;
public:
    virtual void showReplicationStatistics();
    virtual void showSimulationStatistics();
    virtual void showSimulationResponses();
    virtual void showSimulationControls();
private:
    ModelSimulation* _simulation;
    Model* _model;
private:
    List<ModelElement*>* _statsCountersSimulation;
```

```
private:
    const unsigned short _w = 12;
    const unsigned short _nameW = 40;
};
//namespace\\}
#endif /* SIMULATIONREPORTERDEFAULTIMPL1_H */
```

8.307 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationResponse.cpp File Reference

`#include "SimulationResponse.h"`

Include dependency graph for SimulationResponse.cpp:

8.308 SimulationResponse.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationResponse.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 10 de Outubro de 2018, 16:18
 */

#include "SimulationResponse.h"
//#include "AnalysisOfVariance_if.h"

//using namespace GenesysKernel;

SimulationResponse::SimulationResponse(std::string type, std::string name, GetterMember getterMember,
    std::string parent) {
    _type = type;
    _parent = parent;
    _name = name;
    _getterMemberFunction = getterMember;
}

std::string SimulationResponse::show() {
    return "name=" + this->_name + ", type=" + this->_type;
}

std::string SimulationResponse::getName() const {
    return _name;
}

void SimulationResponse::setName(std::string name) {
    _name = name;
}

std::string SimulationResponse::getType() const {
    return _type;
}

std::string SimulationResponse::getParent() const {
```

```
    return _parent;
}

double SimulationResponse::getValue() {
    return this->_getterMemberFunction();
}
```

8.309 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationResponse.h File Reference

```
#include <string>
#include "DefineGetterSetter.h"
```

Include dependency graph for SimulationResponse.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationResponse

8.310 SimulationResponse.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationResponse.h
 * Author: rafael.luiz.cancian
 *
 * Created on 10 de Outubro de 2018, 16:18
 */

#ifndef SIMULATIONRESPONSE_H
#define SIMULATIONRESPONSE_H

#include <string>
#include "DefineGetterSetter.h"

//namespace GenesysKernel {

class SimulationResponse {
public:
```

```
    SimulationResponse(std::string type, std::string name, GetterMember getterMember, std::string parent = "");
    virtual ~SimulationResponse() = default;
public:
    std::string show();
public:
    double getValue();
    std::string getName() const;
    void setName(std::string name);
    std::string getType() const;
    std::string getParent() const;
protected:
    std::string _type;
    std::string _parent;
    std::string _name;
    GetterMember _getterMemberFunction;
};
//namespace\\}
#endif /* SIMULATIONRESPONSE_H */
```

8.311 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationScenario.cpp File Reference

`#include "SimulationScenario.h"`

Include dependency graph for SimulationScenario.cpp:

8.312 SimulationScenario.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SimulationScenario.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 10 de Outubro de 2018, 18:21
 */

#include "SimulationScenario.h"

SimulationScenario::SimulationScenario() = default;

bool SimulationScenario::startSimulation(Simulator *sim, std::string* errorMessage) {
    // \TODO: To implement!
    // *Simulator??????s
    *errorMessage = ""; // just clean the error message
```

```
    return false;
}

void SimulationScenario::setScenarioName(std::string _name) {
    this->_scenarioName = _name;
}

std::string SimulationScenario::getScenarioName() const {
    return _scenarioName;
}

void SimulationScenario::setModelFilename(std::string _modelFilename) {
    this->_modelFilename = _modelFilename;
}

std::string SimulationScenario::getModelFilename() const {
    return _modelFilename;
}

std::list<std::string>* SimulationScenario::getSelectedResponses() const {
    return _selectedResponses;
}

std::list<std::string>* SimulationScenario::getSelectedControls() const {
    return _selectedControls;
}

void SimulationScenario::setScenarioDescription(std::string _scenarioDescription) {
    this->_scenarioDescription = _scenarioDescription;
}

std::string SimulationScenario::getScenarioDescription() const {
    return _scenarioDescription;
}

std::list<std::pair<std::string, double> *> *SimulationScenario::getResponseValues() const {
    return _responseValues;
}

std::list<std::pair<std::string, double> *> *SimulationScenario::getControlValues() const {
    return _controlValues;
}

double SimulationScenario::getResponseValue(const std::string& responseName) {
    for (auto const& r : *_responseValues) {
        if (r->first == responseName) {
            return r->second;
        }
    }
    throw std::invalid_argument("Response does not exist");
}

double SimulationScenario::getControlValue(const std::string& controlName) {
    for (auto const& r : *_controlValues) {
        if (r->first == controlName) {
            return r->second;
        }
    }
    throw std::invalid_argument("Control does not exist");
}

void SimulationScenario::setControl(std::string name, double value) const {
    auto aux = new std::pair<std::string, double>(name, value);
    _controlValues->push_back(aux);
}

void SimulationScenario::setSelectedControls(std::list<std::string> *selectedControls) {
    auto* controls = new std::list<std::string>();
    std::copy(selectedControls->begin(), selectedControls->end(), std::back_inserter(*controls));
    _selectedControls = controls;
}
```

8.313 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SimulationScenario.h File Reference

```
#include <string>
#include <list>
#include "Simulator.h"
```

```
#include "SimulationResponse.h"
#include "SimulationControl.h"
```

Include dependency graph for SimulationScenario.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SimulationScenario

8.314 SimulationScenario.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   SimulationScenario.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 10 de Outubro de 2018, 18:21
00012  */
00013
00014 #ifndef SIMULATIONSCENARIO_H
00015 #define SIMULATIONSCENARIO_H
```

```
#include <string>
#include <list>
#include "Simulator.h"
#include "SimulationResponse.h"
#include "SimulationControl.h"

class SimulationScenario {
public:
    SimulationScenario();
    virtual ~SimulationScenario() = default;
public: // results
    bool startSimulation(Simulator *sim, std::string* errorMessage);
    std::list<std::pair<std::string, double>*>* getResponseValues() const;
    std::list<std::pair<std::string, double>*>* getControlValues() const;
    double getResponseValue(const std::string& responseName);
public: // gets and sets
    void setModelFilename(std::string _modelFilename);
    std::string getModelFilename() const;
    void setScenarioName(std::string _name);
    std::string getScenarioName() const;
    void setScenarioDescription(std::string _scenarioDescription);
    std::string getScenarioDescription() const;
    std::list<std::string>* getSelectedControls() const; // access to the list to insert or remove controls
    double getControlValue(const std::string& controlName);
    std::list<std::string>* getSelectedResponses() const; // access to the list to insert or remove responses
    void setSelectedControls(std::list<std::string>* selectedControls);
    void setControl(std::string name, double value) const;
private:
    std::string _scenarioName;
    std::string _scenarioDescription;
    std::string _modelFilename;
    std::list<std::string>* _selectedControls = new std::list<std::string>();
    std::list<std::string>* _selectedResponses = new std::list<std::string>();
    std::list<std::pair<std::string, double>*>* _controlValues = reinterpret_cast<std::list<std::pair<std::string, double> *> *> (new std::list<std::pair<std::string, double»());
    std::list<std::pair<std::string, double>*>* _responseValues{};
};

#endif /* SIMULATIONSCENARIO_H */

```

8.315 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Simulator.cpp File Reference

```
#include "Simulator.h"
#include "LicenceManager.h"
```

Include dependency graph for Simulator.cpp:

Functions

- GenesysSimulator CreateSimulator2 ()

- Simulator * CreateSimulator ()
- void DestroySimulator2 (GenesysSimulator p)
- void DestroySimulator (Simulator *p)

8.315.1 Function Documentation

8.315.1.1 CreateSimulator() `Simulator* CreateSimulator ( )`

Definition at line 22 of file Simulator.cpp.

8.315.1.2 CreateSimulator2() `GenesysSimulator CreateSimulator2 ( )`

Definition at line 18 of file Simulator.cpp.

8.315.1.3 DestroySimulator() `void DestroySimulator ( Simulator * p )`

Definition at line 30 of file Simulator.cpp.

8.315.1.4 DestroySimulator2() `void DestroySimulator2 ( GenesysSimulator p )`

Definition at line 26 of file Simulator.cpp.

8.316 Simulator.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Genesys.cpp
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 12:48
00012  */
00013
00014 #include "Simulator.h"
00015 #include "LicenceManager.h"
00016 //#include "ToolManager.h"
00017
00018 extern "C" GenesysSimulator CreateSimulator2() {
00019     return new /*GenesysKernel::*/Simulator();
00020 }
00021
00022 extern "C" /*GenesysKernel::*/Simulator* CreateSimulator() {
00023     return new /*GenesysKernel::*/Simulator();
00024 }
00025
00026 extern "C" void DestroySimulator2(GenesysSimulator p) {
```

```
    delete p; // Can use a base class or derived class pointer here
}

extern "C" void DestroySimulator(/*GenesysKernel::*/Simulator* p) {
    delete p; // Can use a base class or derived class pointer here
}


//using namespace GenesysKernel;

Simulator::Simulator() {
    // This is the ONLY method in the entire software where std::cout is allowed.
    std::cout « "STARTING " « _name « ", version " « getVersion() « std::endl;
    _licenceManager = new LicenceManager(this);
    _pluginManager = new PluginManager(this);
    _modelManager = new ModelManager(this);
    _traceManager = new TraceManager(this);
    _experimentManager = new ExperimentManager(this);
    std::cout « '|' « '\t' « _licenceManager->showLicence() « std::endl;
    //std::cout « '|' « '\t' « _licenceManager->showActivationCode() « std::endl;
    //std::cout « '|' « '\t' « _licenceManager->showLimits() « std::endl;
}

PluginManager* Simulator::getPlugins() const {
    return _pluginManager;
}

ModelManager* Simulator::getModels() const {
    return _modelManager;
}

TraceManager* Simulator::getTracer() const {
    return _traceManager;
}

ParserManager* Simulator::getParser() const {
    return _parserManager;
}

ExperimentManager* Simulator::getExperimenter() const {
    return _experimentManager;
}

std::string Simulator::getVersion() const {
    return std::to_string(_versionNumber) + " (" + _versionName + ")";
}

unsigned int Simulator::getVersionNumber() const {
    return _versionNumber;
}

std::string Simulator::getName() const {
    return _name;
}

LicenceManager* Simulator::getLicenceManager() const {
    return _licenceManager;
}

```

8.317 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/Simulator.h File Reference

```
#include <string>
#include <iostream>
#include "Model.h"
#include "Plugin.h"
#include "../util/List.h"
#include "LicenceManager.h"
#include "PluginManager.h"
#include "ModelManager.h"
#include "ParserManager.h"
#include "ExperimentManager.h"
```

Include dependency graph for Simulator.h:

Classes

- class Simulator

Typedefs

- typedef Simulator * GenesysSimulator

8.317.1 Typedef Documentation

8.317.1.1 GenesysSimulator `typedef Simulator* GenesysSimulator`

Definition at line 67 of file Simulator.h.

8.318 Simulator.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Genesys.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 12:48
00012  */
00013
00014 #ifndef GENESYS_H
00015 #define GENESYS_H
00016
00017 #include <string>
00018 #include <iostream>
00019 #include "Model.h"
00020 #include "Plugin.h"
00021 #include "../util/List.h"
00022 #include "LicenceManager.h"
00023 #include "PluginManager.h"
00024 #include "ModelManager.h"
00025 //#include "ToolManager.h"
00026 #include "ParserManager.h"
00027 #include "ExperimentManager.h"
```

```
//namespace GenesysKernel {

class Simulator {
    typedef void (*eventHandler)();
public:
    Simulator();
    virtual ~Simulator() = default;
public: // only get
    std::string getVersion() const;
    unsigned int getVersionNumber() const;
    std::string getName() const;
    LicenceManager* getLicenceManager() const;
    PluginManager* getPlugins() const;
    ModelManager* getModels() const;
    TraceManager* getTracer() const;
    ParserManager* getParser() const;
    ExperimentManager* getExperimenter() const;
private:
private: // attributes 1:1 objects
    LicenceManager* _licenceManager;
    PluginManager* _pluginManager;
    ModelManager* _modelManager;
    TraceManager* _traceManager;
    ParserManager* _parserManager;
    ExperimentManager* _experimentManager;

private: // attributes 1:1 native
    const std::string _name = "GenESyS - GENeric and Expansible SYstem Simulator";
    const std::string _versionName = "russel";
    const unsigned int _versionNumber = 210420;
};
//namespace\\}

// passing a "C" class factory function which instantiates the class.
typedef /*GenesysKernel::*/Simulator* GenesysSimulator;
#endif /* GENESYS_H */

```

8.319 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SinkModel↩ Component.cpp File Reference

`#include "SinkModelComponent.h"`

Include dependency graph for SinkModelComponent.cpp:

8.320 SinkModelComponent.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SinkModelComponent.cpp
```

```
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 14:29
 */

#include "SinkModelComponent.h"

//using namespace GenesysKernel;

SinkModelComponent::SinkModelComponent(Model* model, std::string componentTypename, std::string name)
    : ModelComponent(model, componentTypename, name) {
}

bool SinkModelComponent::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        //this->_reportStatistics = std::stoi((*fields->find("collectStatistics")).second) != 0;
    }
    return res;
}

void SinkModelComponent::_initBetweenReplications() {
}

std::map<std::string, std::string>* SinkModelComponent::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    return fields;
}

bool SinkModelComponent::_check(std::string* errorMessage) {
    // \TODO: To implement!
    *errorMessage += "";
    return true;
}
```

8.321 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SinkModel↩Component.h File Reference

`#include "ModelComponent.h"`

Include dependency graph for SinkModelComponent.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SinkModelComponent

8.322 SinkModelComponent.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SinkModelComponent.h
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 14:29
 */

#ifndef SINKMODELCOMPONENT_H
#define SINKMODELCOMPONENT_H

#include "ModelComponent.h"

//namespace GenesysKernel {

class SinkModelComponent : public ModelComponent {
public:
    SinkModelComponent(Model* model, std::string componentTypename, std::string name = "");
    virtual ~SinkModelComponent() = default;
public:
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:
};
//namespace\\}
#endif /* SINKMODELCOMPONENT_H */

```

8.323 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SourceModel↩ Component.cpp File Reference

```
#include "SourceModelComponent.h"
#include "Model.h"
#include "Attribute.h"
```

Include dependency graph for SourceModelComponent.cpp:

8.324 SourceModelComponent.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
```

```
 * File:   SourceModelCOmponent.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:50
 */

#include "SourceModelComponent.h"
#include "Model.h"
#include "Attribute.h"

//using namespace GenesysKernel;

SourceModelComponent::SourceModelComponent(Model* model, std::string componentTypename, std::string name) : ModelComponent(model, componentTypename, name) {
}

std::string SourceModelComponent::show() {
    std::string text = ModelComponent::show() +
            ",entityType=\"" + _entityType->getName() + "\"" +
            ",firstCreation=" + std::to_string(_firstCreation);
    return text;
}

bool SourceModelComponent::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_entitiesPerCreation = LoadField(fields, "entitiesPerCreation", DEFAULT.entitiesPerCreation);
        this->_firstCreation = LoadField(fields, "firstCreation", DEFAULT.firstCreation);
        this->_timeBetweenCreationsExpression = LoadField(fields, "timeBetweenCreations", DEFAULT.timeBetweenCreationsExpression);
        this->_timeBetweenCreationsTimeUnit = LoadField(fields, "timeBetweenCreationsTimeUnit", DEFAULT.timeBetweenCreationsTimeUnit);
        this->_maxCreationsExpression = LoadField(fields, "maxCreations", DEFAULT.maxCreationsExpression);
        std::string entityTypename = LoadField(fields, "EntityType");
        this->_entityType = dynamic_cast<EntityType*>(_parentModel->getElements()->getElement(Util::TypeOf<EntityType>(), entityTypename));
    }
    return res;
}

void SourceModelComponent::_initBetweenReplications() {
    this->_entitiesCreatedSoFar = 0;
}

std::map<std::string, std::string>* SourceModelComponent::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    SaveField(fields, "entitiesPerCreation", _entitiesPerCreation, DEFAULT.entitiesPerCreation);
    SaveField(fields, "firstCreation", _firstCreation, DEFAULT.firstCreation);
    SaveField(fields, "timeBetweenCreations", _timeBetweenCreationsExpression, DEFAULT.timeBetweenCreationsExpression);
    SaveField(fields, "timeBetweenCreationsTimeUnit", _timeBetweenCreationsTimeUnit, DEFAULT.timeBetweenCreationsTimeUnit);
    SaveField(fields, "maxCreations", _maxCreationsExpression, DEFAULT.maxCreationsExpression);
    SaveField(fields, "EntityType", _entityType->getName());
    return fields;
}

bool SourceModelComponent::_check(std::string* errorMessage) {
    /* include attributes needed */
    ElementManager* elements = _parentModel->getElements();
    std::vector<std::string> neededNames = {"Entity.ArrivalTime"};
    std::string neededName;
    for (unsigned int i = 0; i < neededNames.size(); i++) {
        neededName = neededNames[i];
        if (elements->getElement(Util::TypeOf<Attribute>(), neededName) == nullptr) {
            Attribute* attr1 = new Attribute(_parentModel, neededName);
            elements->insert(attr1);
        }
    }
    bool resultAll = true;
    resultAll &= _parentModel->getElements()->check(Util::TypeOf<EntityType>(), _entityType, "entitytype", errorMessage);
    resultAll &= _parentModel->checkExpression(this->_timeBetweenCreationsExpression, "time between creations", errorMessage);
    return resultAll;
}

void SourceModelComponent::setFirstCreation(double _firstCreation) {
    this->_firstCreation = _firstCreation;
}

double SourceModelComponent::getFirstCreation() const {
    return _firstCreation;
}

```

```
//void SourceModelComponent::setCollectStatistics(bool _collectStatistics) {
//    this->_collectStatistics = _collectStatistics;
//}
//
//bool SourceModelComponent::isCollectStatistics() const {
//    return _collectStatistics;
//}

void SourceModelComponent::setEntityType(EntityType* entityType) {
    _entityType = entityType;
}

EntityType* SourceModelComponent::getEntityType() const {
    return _entityType;
}

void SourceModelComponent::setTimeUnit(Util::TimeUnit _timeUnit) {
    this->_timeBetweenCreationsTimeUnit = _timeUnit;
}

Util::TimeUnit SourceModelComponent::getTimeUnit() const {
    return this->_timeBetweenCreationsTimeUnit;
}

void SourceModelComponent::setTimeBetweenCreationsExpression(std::string _timeBetweenCreations) {
    this->_timeBetweenCreationsExpression = _timeBetweenCreations;
}

std::string SourceModelComponent::getTimeBetweenCreationsExpression() const {
    return _timeBetweenCreationsExpression;
}

void SourceModelComponent::setMaxCreations(std::string _maxCreationsExpression) {
    this->_maxCreationsExpression = _maxCreationsExpression;
}

void SourceModelComponent::setMaxCreations(unsigned long _maxCreations) {
    this->_maxCreationsExpression = std::to_string(_maxCreations);
}

std::string SourceModelComponent::getMaxCreations() const {
    return _maxCreationsExpression;
}

unsigned int SourceModelComponent::getEntitiesCreated() const {
    return _entitiesCreatedSoFar;
}

void SourceModelComponent::setEntitiesCreated(unsigned int _entitiesCreated) {
    this->_entitiesCreatedSoFar = _entitiesCreated;
}

void SourceModelComponent::setEntitiesPerCreation(unsigned int _entitiesPerCreation) {
    this->_entitiesPerCreation = _entitiesPerCreation;
    //this->_entitiesCreatedSoFar = _entitiesPerCreation; // that's because "entitiesPerCreation" entities are included in the future events list BEFORE replication starts (to initialize events list)
}

unsigned int SourceModelComponent::getEntitiesPerCreation() const {
    return _entitiesPerCreation;
}
```

8.325 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/SourceModelComponent.h File Reference

```
#include "ModelComponent.h"
#include <string>
#include <limits>
```

Include dependency graph for SourceModelComponent.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SourceModelComponent
- struct SourceModelComponent::DEFAULT_VALUES

8.326 SourceModelComponent.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SourceModelCOmponent.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:50
 */

#ifndef SOURCEMODELCOMPONENT_H
#define SOURCEMODELCOMPONENT_H

#include "ModelComponent.h"
#include <string>
#include <limits>
//namespace GenesysKernel {

class SourceModelComponent : public ModelComponent {
public:
    SourceModelComponent(Model* model, std::string componentTypename, std::string name = "");
    virtual ~SourceModelComponent() = default;
public: // get & set
    void setFirstCreation(double _firstCreation);
    double getFirstCreation() const;
    void setEntityType(EntityType* _entityType);
    EntityType* getEntityType() const;
    void setTimeUnit(Util::TimeUnit _timeUnit);
    Util::TimeUnit getTimeUnit() const;
    void setTimeBetweenCreationsExpression(std::string _timeBetweenCreations);
    std::string getTimeBetweenCreationsExpression() const;
    void setMaxCreations(unsigned long _maxCreations);
    void setMaxCreations(std::string _maxCreationsExpression);
    std::string getMaxCreations() const;
    unsigned int getEntitiesCreated() const;
    void setEntitiesCreated(unsigned int _entitiesCreated);
    void setEntitiesPerCreation(unsigned int _entitiesPerCreation);
    unsigned int getEntitiesPerCreation() const;
public:
```

```
    virtual std::string show();
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
protected: // get & set
    EntityType* _entityType;

    const struct DEFAULT_VALUES {
        const double firstCreation = 0.0;
        const unsigned int entitiesPerCreation = 1;
        const std::string maxCreationsExpression = std::to_string(std::numeric_limits<unsigned int>::max());
        const std::string timeBetweenCreationsExpression = "EXPO(1.0)";
        const Util::TimeUnit timeBetweenCreationsTimeUnit = Util::TimeUnit::second;
    } DEFAULT;
    double _firstCreation = DEFAULT.firstCreation;
    unsigned int _entitiesPerCreation = DEFAULT.entitiesPerCreation;
    std::string _maxCreationsExpression = DEFAULT.maxCreationsExpression;
    std::string _timeBetweenCreationsExpression = DEFAULT.timeBetweenCreationsExpression;
    Util::TimeUnit _timeBetweenCreationsTimeUnit = DEFAULT.timeBetweenCreationsTimeUnit;
    unsigned int _entitiesCreatedSoFar = 0;
};
//namespace\\}
#endif /* SOURCEMODELCOMPONENT_H */
```

8.327 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/StatisticsCollector.cpp File Reference

```
#include "StatisticsCollector.h"
#include "../TraitsKernel.h"
```

Include dependency graph for StatisticsCollector.cpp:

Typedefs

- typedef TraitsKernel< ModelComponent >::StatisticsCollector_StatisticsImplementation StatisticsClass

8.327.1 Typedef Documentation

8.327.1.1 StatisticsClass `typedef TraitsKernel<ModelComponent>::StatisticsCollector_Statistics↩ Implementation StatisticsClass`

Definition at line 20 of file StatisticsCollector.cpp.

8.328 StatisticsCollector.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   StatisticsCollector.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 30 de Agosto de 2018, 17:24
 */

#include "StatisticsCollector.h"
#include "../TraitsKernel.h"

//using namespace GenesysKernel;


typedef TraitsKernel<ModelComponent>::StatisticsCollector_StatisticsImplementation StatisticsClass;

StatisticsCollector::StatisticsCollector(Model* model, std::string name, ModelElement* parent, bool insertIntoModel) : ModelElement(model, Util::TypeOf<StatisticsCollector>(), name, insertIntoModel) {
    _parent = parent;
    _initStaticsAndCollector();
    _addSimulationResponses();
}

void StatisticsCollector::_addSimulationResponses() {
    GetterMember getterMemberAverage = DefineGetterMember<StatisticsClass>(static_cast<StatisticsClass*> (this->_statistics), &StatisticsClass::average);
    SimulationResponse* resp = new SimulationResponse(Util::TypeOf<StatisticsClass>(), getName() + ".average", getterMemberAverage, getName());
    _parentModel->getResponses()->insert(resp);
    // add the halfwidth as response
    GetterMember getterMemberHalfWidth = DefineGetterMember<StatisticsClass>(static_cast<StatisticsClass*> (this->_statistics), &StatisticsClass::halfWidthConfidenceInterval);
    resp = new SimulationResponse(Util::TypeOf<StatisticsClass>(), /*parentName + ":" + */getName() + ".halfWidth", getterMemberHalfWidth, getName());
    _parentModel->getResponses()->insert(resp);
}

void StatisticsCollector::_initStaticsAndCollector() {
    Collector_if* collector = new TraitsKernel<ModelComponent>::StatisticsCollector_CollectorImplementation();
    _statistics = new StatisticsClass(collector);
    //TraitsKernel<ModelComponent>::StatisticsCollector_StatisticsImplementation(collector);
}

std::string StatisticsCollector::show() {
    std::string parentStr = "";
    if (_parent != nullptr) {
        try {
            parentStr = _parent->getName();
        } catch (...) { // if parent changed or deleted, can cause seg fault
            parentStr = "«INCONSISTENT»"; /* \todo: ++*/
        }
    }
    return ModelElement::show() +
            ",parent=\"" + parentStr + "\"" +
            ",numElements=" + std::to_string(_statistics->numElements());
}

ModelElement* StatisticsCollector::getParent() const {
    return _parent;
}

Statistics_if* StatisticsCollector::getStatistics() const {
    return _statistics;
}

PluginInformation* StatisticsCollector::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<StatisticsCollector>(), &StatisticsCollector::LoadInstance);
    info->setGenerateReport(true);
    return info;
}

ModelElement* StatisticsCollector::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    StatisticsCollector* newElement = new StatisticsCollector(model);
    try {
        newElement->_loadInstance(fields);
```

```
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool StatisticsCollector::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
    }
    return res;
}

std::map<std::string, std::string>* StatisticsCollector::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<StatisticsCollector>());
    std::string parentId = "", parentTypename = "";
    if (this->_parent != nullptr) {
        parentId = std::to_string(_parent->getId());
        parentTypename = _parent->getClassname();
    }
    SaveField(fields, "parentTypename", parentTypename);
    SaveField(fields, "parentId", parentId);
    return fields;
}

bool StatisticsCollector::_check(std::string* errorMessage) {
    // \TODO: To implement!
    *errorMessage += "";
    return true;
}
```

8.329 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/StatisticsCollector.h File Reference

```
#include "ModelElement.h"
#include "../statistics/Statistics_if.h"
#include "ElementManager.h"
#include "Plugin.h"
```

Include dependency graph for StatisticsCollector.h:

This graph shows which files directly or indirectly include this file:

Classes

- class StatisticsCollector

8.330 StatisticsCollector.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   StatisticsCollector.h
 * Author: rafael.luiz.cancian
 *
 * Created on 30 de Agosto de 2018, 17:24
 */

#ifndef STATISTICSCOLLECTOR_H
#define STATISTICSCOLLECTOR_H

#include "ModelElement.h"
#include "../statistics/Statistics_if.h"
#include "ElementManager.h"
#include "Plugin.h"

//namespace GenesysKernel {

class StatisticsCollector : public ModelElement {//, public Statistics_if {
public:
    StatisticsCollector(Model* model, std::string name = "", ModelElement* parent = nullptr, bool insertIntoModel = true);
    virtual ~StatisticsCollector() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    ModelElement* getParent() const;
    Statistics_if* getStatistics() const;

protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
protected:
    void _addSimulationResponses();
private:
    void _initStaticsAndCollector();
private:
    ModelElement* _parent;
    Statistics_if* _statistics; //= new Traits<ModelComponent>::StatisticsCollector_StatisticsImplementation();
};
//namespace\\}
#endif /* STATISTICSCOLLECTOR_H */

```

8.331 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/TraceManager.cpp File Reference

```
#include "TraceManager.h"
#include "../TraitsKernel.h"
```

Include dependency graph for TraceManager.cpp:

8.332 TraceManager.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
```

```
 * and open the template in the editor.
 */

/*
 * File:   TraceManager.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 7 de Novembro de 2018, 11:59
 */

#include "TraceManager.h"
#include "../TraitsKernel.h"

//using namespace GenesysKernel;

TraceManager::TraceManager(Simulator* simulator) {//(Model* model) {
    _simulator = simulator;
    _traceLevel = TraitsKernel<Model>::traceLevel;
}

void TraceManager::setTraceLevel(Util::TraceLevel _traceLevel) {
    this->_traceLevel = _traceLevel;
}

Util::TraceLevel TraceManager::getTraceLevel() const {
    return _traceLevel;
}

Simulator* TraceManager::getParentSimulator() const {
    return _simulator;
}

void TraceManager::addTraceHandler(traceListener traceListener) {
    this->_traceHandlers->insert(traceListener);
}

void TraceManager::addTraceSimulationHandler(traceSimulationListener traceSimulationListener) {
    this->_traceSimulationHandlers->insert(traceSimulationListener);
}

void TraceManager::addTraceErrorHandler(traceErrorListener traceErrorListener) {
    this->_traceErrorHandlers->insert(traceErrorListener);
}

void TraceManager::addTraceReportHandler(traceListener traceReportListener) {
    this->_traceReportHandlers->insert(traceReportListener);
}

void TraceManager::trace(Util::TraceLevel level, std::string text) {
    trace(text, level);
}

void TraceManager::trace(std::string text, Util::TraceLevel level) {
    if (_traceConditionPassed(level)) {
        text = Util::Indent() + text;
        //text = "L" + std::to_string(static_cast<int> (level)) + "    " + Util::Indent() + text;
        TraceEvent e = TraceEvent(text, level);
        /*  \todo:--: somewhere in future it should be interesting to use "auto" and c++17 at least */
        for (std::list<traceListener>::iterator it = this->_traceHandlers->list()->begin(); it !=
 _traceHandlers->list()->end(); it++) {
            (*it)(e);
        }
        for (std::list<traceListenerMethod>::iterator it =
 this->_traceHandlersMethod->list()->begin(); it != _traceHandlersMethod->list()->end(); it++) {
            (*it)(e);
        }

    }
}

void TraceManager::traceError(std::exception e, std::string text) {
    traceError(text, e);
}

void TraceManager::traceError(std::string text, std::exception e) {
    text = Util::Indent() + text;
    TraceErrorEvent exceptEvent = TraceErrorEvent(text, e);
    /*  \todo:--: somewhere in future it should be interesting to use "auto" and c++17 at least */
    for (std::list<traceErrorListener>::iterator it = this->_traceErrorHandlers->list()->begin(); it
 != _traceErrorHandlers->list()->end(); it++) {
        (*it)(exceptEvent);
    }
    for (std::list<traceErrorListenerMethod>::iterator it =
 this->_traceErrorHandlersMethod->list()->begin(); it != _traceErrorHandlersMethod->list()->end();
 it++) {
        (*it)(exceptEvent);
    }
```

```
}

void TraceManager::traceSimulation(Util::TraceLevel level, double time, Entity* entity, ModelComponent* component, std::string text) {
    traceSimulation(time, entity, component, text, level);
}

void TraceManager::traceSimulation(double time, Entity* entity, ModelComponent* component, std::string text, Util::TraceLevel level) {
    if (_traceConditionPassed(level)) {
        text = Util::Indent() + text;
        TraceSimulationEvent e = TraceSimulationEvent(level, time, entity, component, text);
        for (std::list<traceSimulationListener>::iterator it = this->_traceSimulationHandlers->list()->begin(); it != _traceSimulationHandlers->list()->end(); it++) {
            (*it)(e);
        }
        for (std::list<traceSimulationListenerMethod>::iterator it = this->_traceSimulationHandlersMethod->list()->begin(); it != _traceSimulationHandlersMethod->list()->end(); it++) {
            (*it)(e);
        }
    }
}

void TraceManager::traceReport(Util::TraceLevel level, std::string text) {
    traceReport(text, level);
}

void TraceManager::traceReport(std::string text, Util::TraceLevel level) {
    if (_traceConditionPassed(level)) {
        text = Util::Indent() + text;
        TraceEvent e = TraceEvent(text, level);
        for (std::list<traceListener>::iterator it = this->_traceReportHandlers->list()->begin(); it != _traceReportHandlers->list()->end(); it++) {
            (*it)(e);
        }
        for (std::list<traceListenerMethod>::iterator it = this->_traceReportHandlersMethod->list()->begin(); it != _traceReportHandlersMethod->list()->end(); it++) {
            (*it)(e);
        }
    }
}

List<std::string>* TraceManager::errorMessages() const {
    return _errorMessages;
}

bool TraceManager::_traceConditionPassed(Util::TraceLevel level) {
    return /*this->_debugged &&*/ static_cast<int> (this->_traceLevel) >= static_cast<int> (level);
}
```

8.333 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/simulator/TraceManager.h File Reference

```
#include "../util/List.h"
#include <functional>
```

Include dependency graph for TraceManager.h:

This graph shows which files directly or indirectly include this file:

Classes

- class TraceEvent
- class TraceErrorEvent
- class TraceSimulationEvent
- class TraceSimulationProcess
- class TraceManager

Typedefs

- typedef void(∗ traceListener) (TraceEvent)
- typedef void(∗ traceErrorListener) (TraceErrorEvent)
- typedef void(∗ traceSimulationListener) (TraceSimulationEvent)
- typedef void(∗ traceSimulationProcessListener) (TraceSimulationProcess)
- typedef std::function< void(TraceEvent) > traceListenerMethod
- typedef std::function< void(TraceErrorEvent) > traceErrorListenerMethod
- typedef std::function< void(TraceSimulationEvent) > traceSimulationListenerMethod
- typedef std::function< void(TraceSimulationProcess) > traceSimulationProcessListenerMethod

8.333.1 Typedef Documentation

8.333.1.1 traceErrorListener `typedef void(* traceErrorListener) (TraceErrorEvent)`

Definition at line 93 of file TraceManager.h.

8.333.1.2 traceErrorListenerMethod `typedef std::function<void(TraceErrorEvent) > traceErrorListenerMethod`

Definition at line 98 of file TraceManager.h.

8.333.1.3 traceListener `typedef void(* traceListener) (TraceEvent)`

Definition at line 92 of file TraceManager.h.

8.333.1.4 traceListenerMethod `typedef std::function<void(TraceEvent) > traceListenerMethod`

Definition at line 97 of file TraceManager.h.

8.333.1.5 traceSimulationListener `typedef void(* traceSimulationListener) (TraceSimulationEvent)`

Definition at line 94 of file TraceManager.h.

8.333.1.6 traceSimulationListenerMethod `typedef std::function<void(TraceSimulationEvent) > traceSimulationListenerMethod`

Definition at line 99 of file TraceManager.h.

8.333.1.7 traceSimulationProcessListener `typedef void(* traceSimulationProcessListener) (TraceSimulationProcess`

Definition at line 95 of file TraceManager.h.

8.333.1.8 traceSimulationProcessListenerMethod `typedef std::function<void(TraceSimulationProcess) > traceSimulationProcessListenerMethod`

Definition at line 100 of file TraceManager.h.

8.334 TraceManager.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   TraceManager.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 7 de Novembro de 2018, 11:59
00012  */
00013
00014 #ifndef TRACEMANAGER_H
00015 #define TRACEMANAGER_H
00016
00017 #include "../util/List.h"
00018 #include <functional>
00019 //namespace GenesysKernel {
00020 class Model;
00021 class Entity;
00022 class ModelComponent;
00023 class Simulator;
00024
00025 class TraceEvent {
00026 public:
00027     TraceEvent(Util::TraceLevel level, std::string text) {
00028         _tracelevel = level;
00029         _text = text;
00030     }
00031     TraceEvent(std::string text, Util::TraceLevel level = Util::TraceLevel::L8_detailed) {
```

```
        _tracelevel = level;
        _text = text;
    }
    Util::TraceLevel getTracelevel() const {
        return _tracelevel;
    }
    std::string getText() const {
        return _text;
    }
private:
    Util::TraceLevel _tracelevel;
    std::string _text;
};

class TraceErrorEvent : public TraceEvent {
public:
    TraceErrorEvent(std::string text, std::exception e) : TraceEvent(text, Util::TraceLevel::L3_errorRecover) {
        _e = e;
    }
    std::exception getException() const {
        return _e;
    }
private:
    std::exception _e;
};

class TraceSimulationEvent : public TraceEvent {
public:
    TraceSimulationEvent(Util::TraceLevel level, double time, Entity* entity, ModelComponent* component, std::string text) : TraceEvent(text, level) {
        _time = time;
        _entity = entity;
        _component = component;
    }
    ModelComponent* getComponent() const {
        return _component;
    }
    Entity* getEntity() const {
        return _entity;
    }
    double getTime() const {
        return _time;
    }

private:
    double _time;
    Entity* _entity;
    ModelComponent* _component;
};

class TraceSimulationProcess : public TraceEvent {
public:
    TraceSimulationProcess(std::string text, Util::TraceLevel level = Util::TraceLevel::L8_detailed) : TraceEvent(text, level) {
    }
};

// for handlers that are simple functions
typedef void (*traceListener)(TraceEvent);
typedef void (*traceErrorListener)(TraceErrorEvent);
typedef void (*traceSimulationListener)(TraceSimulationEvent);
typedef void (*traceSimulationProcessListener)(TraceSimulationProcess);
// for handlers that are class members (methods)
typedef std::function<void(TraceEvent) > traceListenerMethod;
typedef std::function<void(TraceErrorEvent) > traceErrorListenerMethod;
typedef std::function<void(TraceSimulationEvent) > traceSimulationListenerMethod;
typedef std::function<void(TraceSimulationProcess) > traceSimulationProcessListenerMethod;

class TraceManager {
public:
    TraceManager(Simulator* simulator); //(Model* model);
    virtual ~TraceManager() = default;
public: // add trace handlers
    // for handlers that are simple functions
    void addTraceHandler(traceListener traceListener);
    void addTraceReportHandler(traceListener traceReportListener);
    void addTraceSimulationHandler(traceSimulationListener traceSimulationListener);
    void addTraceErrorHandler(traceErrorListener traceErrorListener);
    // for handlers that are class members (methods)
    template<typename Class> void addTraceHandler(Class * object, void (Class::*function)(TraceEvent));
    template<typename Class> void addTraceErrorHandler(Class * object, void (Class::*function)(TraceErrorEvent));
    template<typename Class> void addTraceReportHandler(Class * object, void (Class::*function)(TraceEvent));
    template<typename Class> void addTraceSimulationHandler(Class * object, void
```

```
(Class::*function)(TraceSimulationEvent));
public: // traces (invoke trace handlers)
    void trace(Util::TraceLevel level, std::string text);
    void traceError(std::exception e, std::string text);
    void traceReport(Util::TraceLevel level, std::string text);
    void traceSimulation(Util::TraceLevel level, double time, Entity* entity, ModelComponent*
 component, std::string text);
public: // traces (invoke trace handlers) SINCE 20191025 NEW TRACES JUST INVERTED THE PARAMETERS,
 MAKING TRACELEVEL OPTIONAL
    void trace(std::string text, Util::TraceLevel level = Util::TraceLevel::L8_detailed);
    void traceError(std::string text, std::exception e);
    void traceReport(std::string text, Util::TraceLevel level = Util::TraceLevel::L2_results);
    void traceSimulation(double time, Entity* entity, ModelComponent* component, std::string text,
 Util::TraceLevel level = Util::TraceLevel::L8_detailed);
public:
    List<std::string>* errorMessages() const;
    void setTraceLevel(Util::TraceLevel _traceLevel);
    Util::TraceLevel getTraceLevel() const;
    Simulator* getParentSimulator() const;
private:
    //void _addHandler(List<traceListener>* list, )
    bool _traceConditionPassed(Util::TraceLevel level);
private: // trace listener
    // for handlers that are simple functions
    List<traceListener>* _traceHandlers = new List<traceListener>();
    List<traceErrorListener>* _traceErrorHandlers = new List<traceErrorListener>();
    List<traceListener>* _traceReportHandlers = new List<traceListener>();
    List<traceSimulationListener>* _traceSimulationHandlers = new List<traceSimulationListener>();
    // for handlers that are class members (methods)
    List<traceListenerMethod>* _traceHandlersMethod = new List<traceListenerMethod>();
    List<traceErrorListenerMethod>* _traceErrorHandlersMethod = new List<traceErrorListenerMethod>();
    List<traceListenerMethod>* _traceReportHandlersMethod = new List<traceListenerMethod>();
    List<traceSimulationListenerMethod>* _traceSimulationHandlersMethod = new
 List<traceSimulationListenerMethod>();
private:
    //Model* _model;
    Simulator* _simulator;
private:
    Util::TraceLevel _traceLevel; // = Util::TraceLevel::L9_mostDetailed;
    double _lastTimeTraceSimulation = -1.0;
    Util::identification _lastEntityTraceSimulation = 0;
    Util::identification _lastModuleTraceSimulation = 0;
    List<std::string>* _errorMessages; /* \todo: 18/08/24 this is a new one. several methods should
 use it */

};

// implementation for template methods
template<typename Class> void TraceManager::addTraceHandler(Class * object, void
 (Class::*function)(TraceEvent)) {
    this->_traceHandlersMethod->insert(std::bind(function, object, std::placeholders::_1));
}
template<typename Class> void TraceManager::addTraceErrorHandler(Class * object, void
 (Class::*function)(TraceErrorEvent)) {
    this->_traceErrorHandlersMethod->insert(std::bind(function, object, std::placeholders::_1));
}
template<typename Class> void TraceManager::addTraceReportHandler(Class * object, void
 (Class::*function)(TraceEvent)) {
    this->_traceReportHandlersMethod->insert(std::bind(function, object, std::placeholders::_1));
}
template<typename Class> void TraceManager::addTraceSimulationHandler(Class * object, void
 (Class::*function)(TraceSimulationEvent)) {
    this->_traceSimulationHandlersMethod->insert(std::bind(function, object, std::placeholders::_1));
}
//namespace\\}
#endif /* TRACEMANAGER_H */
```

8.335 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Collector_if.h File Reference

```
#include <string>
#include <functional>
```

Include dependency graph for Collector_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Collector_if

Typedefs

- typedef std::function< void(double) > CollectorAddValueHandler
- typedef std::function< void() > CollectorClearHandler

Functions

- template<typename Class >
  CollectorAddValueHandler setCollectorAddValueHandler (void(Class::∗function)(double), Class ∗object)
- template<typename Class >
  CollectorClearHandler setCollectorClearHandler (void(Class::∗function)(), Class ∗object)

8.335.1 Typedef Documentation

8.335.1.1 CollectorAddValueHandler `typedef std::function<void(double) > CollectorAddValueHandler`

Definition at line 22 of file Collector_if.h.

8.335.1.2 CollectorClearHandler `typedef std::function<void() > CollectorClearHandler`

Definition at line 28 of file Collector_if.h.

8.335.2 Function Documentation

8.335.2.1 setCollectorAddValueHandler()

```
template<typename Class >
CollectorAddValueHandler setCollectorAddValueHandler (
            void(Class::*)(double) function,
            Class * object )
```

Definition at line 24 of file Collector_if.h.

Here is the caller graph for this function:

8.335.2.2 setCollectorClearHandler()

```
template<typename Class >
CollectorClearHandler setCollectorClearHandler (
            void(Class::*)() function,
            Class * object )
```

Definition at line 30 of file Collector_if.h.

Here is the caller graph for this function:

8.336 Collector_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Collector_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 14:16
 */

#ifndef COLLECTOR_IF_H
#define COLLECTOR_IF_H

#include <string>
#include <functional>

typedef std::function<void(double) > CollectorAddValueHandler;
template<typename Class>
CollectorAddValueHandler setCollectorAddValueHandler(void (Class::*function)(double), Class * object) {
    return std::bind(function, object, std::placeholders::_1);
}

typedef std::function<void() > CollectorClearHandler;
template<typename Class>
CollectorClearHandler setCollectorClearHandler(void (Class::*function)(), Class * object) {
    return std::bind(function, object);
}

class Collector_if {
public:
    virtual void clear() = 0;
    virtual void addValue(double value) = 0;
    virtual double getLastValue() = 0;
    virtual unsigned long numElements() = 0;
public:
    virtual void setAddValueHandler(CollectorAddValueHandler addValueHandler) = 0;
    virtual void setClearHandler(CollectorClearHandler clearHandler) = 0;
};

#endif /* COLLECTOR_IF_H */
```

8.337 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafile_if.h File Reference

```
#include "Collector_if.h"
```

Include dependency graph for CollectorDatafile_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class CollectorDatafile_if

8.338 CollectorDatafile_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CollectorDatafile_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 30 de Agosto de 2018, 16:45
 */

#ifndef COLLECTORDATAFILE_IF_H
#define COLLECTORDATAFILE_IF_H

#include "Collector_if.h"

class CollectorDatafile_if : public Collector_if {
public:
    virtual double getValue(unsigned int rank) = 0;
    virtual void seekFirstValue() = 0;
    virtual double getNextValue() = 0;
    virtual std::string getDataFilename() = 0;
    virtual void setDataFilename(std::string filename) = 0;
};

#endif /* COLLECTORDATAFILE_IF_H */
```

8.339 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafile↩ DefaultImpl1.cpp File Reference

```
#include "CollectorDatafileDefaultImpl1.h"
#include <fstream>
#include <iostream>
#include <iomanip>
#include <limits>
#include <sstream>
```

Include dependency graph for CollectorDatafileDefaultImpl1.cpp:

8.340 CollectorDatafileDefaultImpl1.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   CollectorDatafileDefaultImpl11.cpp
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 1 de Agosto de 2018, 20:58
00012  */
00013
00014 #include "CollectorDatafileDefaultImpl1.h"
00015 #include <fstream>
00016 #include <iostream>
00017 #include <iomanip>
00018 #include <limits>
00019 #include <sstream>
00020
00021 CollectorDatafileDefaultImpl1::CollectorDatafileDefaultImpl1() {
00022 }
00023
00024 void CollectorDatafileDefaultImpl1::clear() {
00025     std::ofstream file;
00026     try {
00027         file.open(_filename, std::ofstream::out | std::ofstream::trunc);
00028         file.close();
00029     } catch (const std::exception& e) {
00030         throw "ERROR - can't open the file ";
00031     }
00032     _numElements = 0;
00033     _lastValaue = NULL; //\TODO: Change it to NaN or other default/null value
00034 }
00035
00036 void CollectorDatafileDefaultImpl1::addValue(double value) {
```

```
    std::ofstream file;
    try {
        if (this->_numElements > 0) {
            file.open(_filename, std::ofstream::out | std::ofstream::app);
        } else {
            file.open(_filename, std::ofstream::out | std::ofstream::trunc);
        }
        file.write(reinterpret_cast<char*> (&value), sizeof (double));
        file.close();
    } catch (const std::exception& e) {
        throw "ERROR - can't open the file ";
    }
    _numElements++;
    _lastValaue = value;
}

double CollectorDatafileDefaultImpl1::getLastValue() {
    return _lastValaue;
}

unsigned long CollectorDatafileDefaultImpl1::numElements() {
    return _numElements;
}

double CollectorDatafileDefaultImpl1::getValue(unsigned int num) {
    std::ifstream file;
    double value;
    if (num > _numElements) {
        throw "ERROR - num greater than numElements";
    }
    try {
        file.open(_filename, std::ifstream::binary | std::ifstream::in);
        file.seekg(sizeof (double) * num);
        //valueType d;
        file.read(reinterpret_cast<char*> (&value), sizeof (double));
        file.close();
    } catch (const std::exception& e) {
        throw "ERROR - can't open the file or get the line ";
    }
    return value;
}

double CollectorDatafileDefaultImpl1::getNextValue() {
    return 0.0; // \todo:
}

void CollectorDatafileDefaultImpl1::seekFirstValue() {
}

std::string CollectorDatafileDefaultImpl1::getDataFilename() {
    return _filename;
}

void CollectorDatafileDefaultImpl1::setDataFilename(std::string filename) {
    _filename = filename;
}

void CollectorDatafileDefaultImpl1::setAddValueHandler(CollectorAddValueHandler addValueHandler) {
    // \todo: just to use it
}

void CollectorDatafileDefaultImpl1::setClearHandler(CollectorClearHandler clearHandler) {
    // \todo: just to use it
}
```

8.341 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDatafileDefaultImpl1.h File Reference

```
#include <string>
#include "CollectorDatafile_if.h"
```

Include dependency graph for CollectorDatafileDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class CollectorDatafileDefaultImpl1

8.342 CollectorDatafileDefaultImpl1.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CollectorDatafileDefaultImpl1.h
 * Author: rafael.luiz.cancian
 *
 * Created on 1 de Agosto de 2018, 20:58
 */

#ifndef COLLECTORDATAFILEDEFAULTIMPL1_H
#define COLLECTORDATAFILEDEFAULTIMPL1_H

#include <string>

#include "CollectorDatafile_if.h"

```

```
class CollectorDatafileDefaultImpl1 : public CollectorDatafile_if {
public:
    CollectorDatafileDefaultImpl1();
    virtual ~CollectorDatafileDefaultImpl1() = default;
public: // inherited from Collector_if
    void clear();
    void addValue(double value);
    double getLastValue();
    unsigned long numElements();
public:
    double getValue(unsigned int num);
    double getNextValue();
    void seekFirstValue();
    std::string getDataFilename();
    void setDataFilename(std::string filename);
public:
    void setAddValueHandler(CollectorAddValueHandler addValueHandler);
    void setClearHandler(CollectorClearHandler clearHandler);
private:
    std::string _filename;
    double _lastValaue;
    unsigned int _numElements;
};

#endif /* COLLECTORDATAFILEDEFAULTIMPL1_H */
```

8.343 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDefaultImpl1.cpp File Reference

`#include "CollectorDefaultImpl1.h"`

Include dependency graph for CollectorDefaultImpl1.cpp:

8.344 CollectorDefaultImpl1.cpp

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CollectorDefaultImpl1.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 1 de Agosto de 2018, 20:43
 */

#include "CollectorDefaultImpl1.h"

//using namespace GenesysKernel;

CollectorDefaultImpl1::CollectorDefaultImpl1() {
}

void CollectorDefaultImpl1::clear() {
    _numElements = 0;
    if (_clearHandler != nullptr) {
        _clearHandler();
    }
}

void CollectorDefaultImpl1::addValue(double value) {
    _lastValue = value;
    _numElements++;
    if (_addValueHandler != nullptr) {
        _addValueHandler(value);
    }
}

double CollectorDefaultImpl1::getLastValue() {
    return this->_lastValue;
}

unsigned long CollectorDefaultImpl1::numElements() {
    return this->_numElements;
}

void CollectorDefaultImpl1::setAddValueHandler(CollectorAddValueHandler addValueHandler) {
    _addValueHandler = addValueHandler;
}

void CollectorDefaultImpl1::setClearHandler(CollectorClearHandler clearHandler) {
    _clearHandler = clearHandler;
}
```

8.345 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/CollectorDefault↩ Impl1.h File Reference

```
#include "Collector_if.h"
```

Include dependency graph for CollectorDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class CollectorDefaultImpl1

8.346 CollectorDefaultImpl1.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CollectorDefaultImpl1.h
 * Author: rafael.luiz.cancian
 *
 * Created on 1 de Agosto de 2018, 20:43
 */

#ifndef COLLECTORDEFAULTIMPL1_H
#define COLLECTORDEFAULTIMPL1_H

#include "Collector_if.h"
//namespace GenesysKernel {

class CollectorDefaultImpl1 : public Collector_if {
public:
    CollectorDefaultImpl1();
    virtual ~CollectorDefaultImpl1() = default;
public:
    void clear();
    void addValue(double value);
    double getLastValue();
    unsigned long numElements();
public:
    void setAddValueHandler(CollectorAddValueHandler addValueHandler);
    void setClearHandler(CollectorClearHandler clearHandler);
```

```
private:
    double _lastValue;
    unsigned long _numElements = 0;
    CollectorAddValueHandler _addValueHandler = nullptr;
    CollectorClearHandler _clearHandler = nullptr;
};
//namespace\\}
#endif /* COLLECTORDEFAULTIMPL1_H */
```

8.347 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Sampler_if.h File Reference

This graph shows which files directly or indirectly include this file:

Classes

- class Sampler_if
- struct Sampler_if::RNG_Parameters

8.348 Sampler_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Sampler_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 13:20
 */

#ifndef Sampler_IF_H
#define Sampler_IF_H

class Sampler_if {
public:

    struct RNG_Parameters {
        virtual ~RNG_Parameters() = default;
    };
public: // RNG
    virtual double random() = 0;
public: // continuous probability distributions
    virtual double sampleBeta(double alpha, double beta, double infLimit, double supLimit) = 0;
    virtual double sampleBeta(double alpha, double beta) = 0;
    virtual double sampleErlang(double mean, int M) = 0;
    virtual double sampleExponential(double mean) = 0;
    virtual double sampleGamma(double mean, double alpha) = 0;
    virtual double sampleGumbell(double mode, double scale) = 0;
```

```
    virtual double sampleLogNormal(double mean, double stddev) = 0;
    virtual double sampleNormal(double mean, double stddev) = 0;
    virtual double sampleTriangular(double min, double mode, double max) = 0;
    virtual double sampleUniform(double min, double max) = 0;
    virtual double sampleWeibull(double alpha, double scale) = 0;
public: // discrete probability distributions
    virtual double sampleBinomial(int trials, double p) = 0;
    virtual double sampleBernoulli(double p) = 0;
    virtual double sampleDiscrete(double acumProb, double value, ...) = 0;
    virtual double sampleDiscrete(double acumProb, double *prob, double *value, int size) = 0;
    virtual double sampleGeometric(double p) = 0;
public:
    virtual void setRNGparameters(RNG_Parameters* param) = 0;
    virtual RNG_Parameters* getRNGparameters() const = 0;
};

#endif /* Sampler_IF_H */

```

8.349 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerBoostImpl.cpp File Reference

`#include "SamplerBoostImpl.h"`

Include dependency graph for SamplerBoostImpl.cpp:

8.350 SamplerBoostImpl.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SamplerBoostImpl.cpp
 * Author: rlcancian
 *
 * Created on 21 de Outubro de 2019, 17:24
 */

#include "SamplerBoostImpl.h"
```

```
//using namespace boost::random;

SamplerBoostImpl::SamplerBoostImpl() {
}

double SamplerBoostImpl::random() {
    //uniform_int_distribution<> dist(0.0, 1.0);
    return 0.0; // dist(_gen);
}

double SamplerBoostImpl::sampleBeta(double alpha, double beta, double infLimit, double supLimit) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleDiscrete(double acumProb, double value, ...) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleErlang(double mean, int M) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleExponential(double mean) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleGamma(double mean, double alpha) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleLogNormal(double mean, double stddev) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleNormal(double mean, double stddev) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleTriangular(double min, double mode, double max) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

double SamplerBoostImpl::sampleUniform(double min, double max) {
    //uniform_int_distribution<> dist(min, max);
    return 0.0; //dist(_gen);
}

double SamplerBoostImpl::sampleWeibull(double alpha, double scale) {
    return 0.0; // \todo: not implemented yet (dummy impl)
}

void SamplerBoostImpl::setRNGparameters(Sampler_if::RNG_Parameters* param) {

}

Sampler_if::RNG_Parameters* SamplerBoostImpl::getRNGparameters() const {
    // \todo: toimplement
}
```

8.351 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerBoostImpl.h File Reference

```
#include "Sampler_if.h"
```

Include dependency graph for SamplerBoostImpl.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SamplerBoostImpl
- struct SamplerBoostImpl::BoostImplRNG_Parameters

8.352 SamplerBoostImpl.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SamplerBoostImpl.h
 * Author: rlcancian
 *
 * Created on 21 de Outubro de 2019, 17:24
```

```
 */

#ifndef SAMPLERBOOSTIMPL_H
#define SAMPLERBOOSTIMPL_H

#include "Sampler_if.h"
//#include <boost/random.hpp>

class SamplerBoostImpl : public Sampler_if {
public:

    struct BoostImplRNG_Parameters : public RNG_Parameters {
        double param;
    };

public:
    SamplerBoostImpl();
    virtual ~SamplerBoostImpl() = default;
public: // probability distributions
    virtual double random();
    virtual double sampleBeta(double alpha, double beta, double infLimit, double supLimit);
    virtual double sampleDiscrete(double acumProb, double value, ...);
    virtual double sampleErlang(double mean, int M);
    virtual double sampleExponential(double mean);
    virtual double sampleGamma(double mean, double alpha);
    virtual double sampleLogNormal(double mean, double stddev);
    virtual double sampleNormal(double mean, double stddev);
    virtual double sampleTriangular(double min, double mode, double max);
    virtual double sampleUniform(double min, double max);
    virtual double sampleWeibull(double alpha, double scale);
public:
    void reset();
public:
    virtual void setRNGparameters(Sampler_if::RNG_Parameters* param);
    virtual RNG_Parameters* getRNGparameters() const;
private:
    //boost::random::mt19937 _gen;
};

#endif /* SAMPLERBOOSTIMPL_H */

```

8.353 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerDefault↩ Impl1.cpp File Reference

```
#include <cmath>
#include <complex>
#include <cassert>
#include <random>
#include "SamplerDefaultImpl1.h"
```

Include dependency graph for SamplerDefaultImpl1.cpp:

8.354 SamplerDefaultImpl1.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SamplerDefaultImpl1.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 2 de Agosto de 2018, 01:10
 * 22/10/2019 old genesys code reinserted
 */

#include <cmath>
#include <complex>
#include <cassert>
#include <random>

#include "SamplerDefaultImpl1.h"

//using namespace GenesysKernel;

SamplerDefaultImpl1::SamplerDefaultImpl1() {
    reset();
}

void SamplerDefaultImpl1::reset() {
    _xi = static_cast<DefaultImpl1RNG_Parameters*> (_param)->seed;
    _normalflag = false;
}

void SamplerDefaultImpl1::setRNGparameters(Sampler_if::RNG_Parameters * param) {

    _param = param; // there is a better solution for this...
}

Sampler_if::RNG_Parameters * SamplerDefaultImpl1::getRNGparameters() const {
    return _param;
}

double SamplerDefaultImpl1::random() {
    unsigned long a = static_cast<DefaultImpl1RNG_Parameters*> (_param)->a;
    unsigned long m = static_cast<DefaultImpl1RNG_Parameters*> (_param)->m;
    _xi = (uint64_t) _xi * a % m;
    //unsigned long c = static_cast<DefaultImpl1RNG_Parameters*> (_param)->c;
    // _xi = ((uint64_t)_xi * a + c) % m;
    return (double) (_xi) / (double) (m);
}

double SamplerDefaultImpl1::sampleUniform(double min, double max) {
    return min + (max - min) * random();
}

double SamplerDefaultImpl1::sampleExponential(double mean) {
    return mean * (-std::log(random()));
}

double SamplerDefaultImpl1::sampleErlang(double mean, int M) {
    int i;
    double P;
    assert((mean >= 0.0) && (M > 0));
    P = 1;
    for (i = 1; i <= M; i++) {
        P *= random();
    }
    return (mean / M) * (-log(P));
}

double SamplerDefaultImpl1::sampleNormal(double mean, double stddev) {
    double z;
    if (_normalflag) {
        z = _lastnormal;
    } else {
        double u1 = random(), u2 = random();
        z = std::sqrt(-2 * std::log(u1)) * std::sin(2 * M_PI * u2);
        _lastnormal = std::sqrt(-2 * std::log(u1)) * std::cos(2 * M_PI * u2);
    }
    _normalflag = !_normalflag;
    return mean + stddev*z;
}

double SamplerDefaultImpl1::sampleLogNormal(double mean, double stddev) {
    double meanNorm, DispNorm;
    //assert(!((mean <= 0.0) || (stddev <= 0.0)));
```

```
    DispNorm = log((stddev * stddev) / (mean * mean) + 1.0);
    meanNorm = log(mean) - 0.5 * DispNorm;
    return exp(sampleNormal(meanNorm, sqrt(DispNorm)));
}

double SamplerDefaultImpl1::sampleTriangular(double min, double mode, double max) {
    double Part1, Part2, Full, R;
    assert(!((min > mode) || (max < mode) || (min > max)));
    Part1 = mode - min;
    Part2 = max - mode;
    Full = max - min;
    R = random();
    if (R <= Part1 / Full)
        return min + sqrt(Part1 * Full * R);
    else
        return max - sqrt(Part2 * Full * (1.0 - R));
}

double SamplerDefaultImpl1::sampleDiscrete(double acumProb, double value, ...) {
    // \todo: to implement
    return 0.0;
}

double SamplerDefaultImpl1::sampleDiscrete(double acumProb, double *prob, double *value, int size) {
    double cdf = 0;
    double x;
    x = random();

    for (int i = 0; i < size; i++) {
        cdf += prob[i] / acumProb;
        if (x <= cdf) {
            return value[i];
        }
    }
    return value[size - 1];
}

/*
double SamplerDefaultImpl1::_gammaJonk(double alpha) {
    double R;
    double R1, R2, X, Y;
    do {
        do {
            R1 = random();
            R2 = random();
        } while (!((R1 > 1e-30) and (R2 > 1e-30)));
        if (log(R2) / alpha < -1e3)
            X = 0;
        else
            X = exp(log(R2) / alpha);
        if ((log(R1) / (1 - alpha) < -1e3))
            Y = 0;
        else
            Y = exp(log(R1) / (1 - alpha));
    } while (!(X + Y <= 1));
    do {
        R = random();
    } while (!(R > 1e-20));
    return -log(R) * X / (Y + X);
}

double SamplerDefaultImpl1::sampleGamma(double mean, double alpha) {
    int i;
    double P;
    int IntAlpha;
    double OstAlpha;
    assert(!((mean <= 0.0) || (alpha <= 0.0)));
    if (alpha < 1.0)
        return (mean / alpha) * _gammaJonk(alpha);
    else {
        if (alpha == 1.0)
            return mean * (-log(random()));
        else {
            IntAlpha = round(alpha);
            OstAlpha = alpha - IntAlpha;
            do {
                P = 1;
                for (i = 1; i <= IntAlpha; i++)
                    P *= random();
            } while (!(P > 0));
            if (OstAlpha > 0)
                return (mean / alpha)*((-log(P)) + _gammaJonk(OstAlpha));
            else
                return (mean / alpha)*(-log(P));
        };
    };
}
```

```
 */

double SamplerDefaultImpl1::sampleGamma(double alpha, double beta) {
    double u, v, w, delta;
    double eps;
    double nt;
    int n;
    n = floor(alpha);
    delta = alpha - n;
    while (1) {
        u = random();
        v = random();
        w = random();
        if (u <= (M_E / (M_E + delta))) {
            eps = pow(v, 1 / delta);
            nt = w * pow(eps, delta - 1);
        } else {
            eps = 1 - log(v);
            nt = w * exp(-eps);
        }
        if (nt > (pow(eps, delta - 1) * exp(-eps))) {
            continue;
        }
        break;
    }
    double gamma_n = 0;
    for (int i = 0; i < n; i++) {
        gamma_n += log(random());
    }
    double gamma = beta * (eps - gamma_n);
    return gamma;
}

double SamplerDefaultImpl1::sampleBeta(double alpha, double beta, double infLimit, double supLimit) {
    double X = sampleBeta(alpha, beta);
    return infLimit + (supLimit - infLimit) * X;
}

double SamplerDefaultImpl1::sampleWeibull(double alpha, double scale) {
    assert(!((alpha <= 0.0) || (scale <= 0.0)));
    return exp(log(scale * (-log(random()))) / alpha);
}

double SamplerDefaultImpl1::sampleBinomial(int trials, double p) {
    double binomial = 0.0;
    double U;

    for (int i = 0; i < trials; i++) {
        U = random();
        if (U < p) {
            binomial += 1.0;
        }
    }

    return binomial;
}

double SamplerDefaultImpl1::sampleBernoulli(double p) {
    double U;

    U = random();
    if (U <= p) {
        return 1.0;
    }
    return 0.0;
}

double SamplerDefaultImpl1::sampleGeometric(double p) {
    assert(p > 0 && p <= 1);
    double rand = random();

    return ceil(log(1 - rand) / log(1 - p));
}

double SamplerDefaultImpl1::sampleGumbell(double mode, double scale) {
    double x;
    x = random();
    return mode - (scale * log(-log(x)));
}

double SamplerDefaultImpl1::sampleBeta(double alpha, double beta) {

    double x = sampleGamma(alpha, 1);
    double y = sampleGamma(beta, 1);
    return x / (x + y);

}
```

```
00260
```

8.355 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SamplerDefaultImpl1.h File Reference

```
#include "Sampler_if.h"
```

Include dependency graph for SamplerDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SamplerDefaultImpl1
- struct SamplerDefaultImpl1::DefaultImpl1RNG_Parameters

8.356 SamplerDefaultImpl1.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   SamplerDefaultImpl1.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 2 de Agosto de 2018, 01:10
00012  */
00013
00014 #ifndef SAMPLERDEFAULTIMPL1_H
00015 #define SAMPLERDEFAULTIMPL1_H
00016
00017 #include "Sampler_if.h"
00018 //namespace GenesysKernel {
00019
00020 class SamplerDefaultImpl1 : public Sampler_if {
00021 public:
```

```
00022
00023     struct DefaultImpl1RNG_Parameters : public RNG_Parameters {
00024         uint32_t seed = 16021974;
00025         uint32_t a = 279470273u; // multiplier
00026         uint32_t m = 0xfffffffb; // module
00027         uint32_t c = 0; // increment
00028         ~DefaultImpl1RNG_Parameters() = default;
00029     };
00030 public:
00031     SamplerDefaultImpl1();
00032     virtual ~SamplerDefaultImpl1() = default;
00033 public: // RNG
00034     virtual double random();
00035 public: // continuous probability distributions
00036     virtual double sampleBeta(double alpha, double beta, double infLimit, double supLimit);
00037     virtual double sampleBeta(double alpha, double beta);
00038     virtual double sampleErlang(double mean, int M);
00039     virtual double sampleExponential(double mean);
00040     virtual double sampleGamma(double mean, double alpha);
00041     virtual double sampleGumbell(double mode, double scale);
00042     virtual double sampleLogNormal(double mean, double stddev);
00043     virtual double sampleNormal(double mean, double stddev);
00044     virtual double sampleTriangular(double min, double mode, double max);
00045     virtual double sampleUniform(double min, double max);
00046     virtual double sampleWeibull(double alpha, double scale);
00047 public: // discrete probability distributions
00048     virtual double sampleBinomial(int trials, double p);
00049     virtual double sampleBernoulli(double p);
00050     virtual double sampleDiscrete(double acumProb, double value, ...);
00051     virtual double sampleDiscrete(double acumProb, double *prob, double *value, int size);
00052     virtual double sampleGeometric(double p);
00053 public:
00054     void reset();
00055 public:
00056     virtual void setRNGparameters(RNG_Parameters* param);
00057     virtual RNG_Parameters* getRNGparameters() const;
00058 private:
00059     RNG_Parameters* _param = new DefaultImpl1RNG_Parameters();
00060     uint32_t _xi;
00061     bool _normalflag;
00062     double _lastnormal;
00063 };
00064 //namespace\\}
00065 #endif /* SAMPLERDEFAULTIMPL1_H */
00066
```

8.357 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SorttFile.cpp File Reference

```
#include "SorttFile.h"
#include <fstream>
#include <iostream>
#include <iomanip>
```

Include dependency graph for SorttFile.cpp:

8.358 SorttFile.cpp

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.cc to edit this template
 */

/*
 * File:   SorttFile.cpp
 * Author: rlcancian
 *
 * Created on 20 de novembro de 2019, 22:27
 */

#include "SorttFile.h"
#include <fstream>
#include <iostream>
#include <iomanip>

SortFile::SortFile() {
}

bool SortFile::sort() {
    run(0, _numElements - 1);
    return true; //???
}

void SortFile::setDataFilename(std::string filename) {
    _filename = filename;
    std::ifstream file;
    file.open(_filename, std::ifstream::binary | std::ifstream::in);
    file.seekg(0, file.end); // changed
    _numElements = (file.tellg() / sizeof (double));

}

void SortFile::run(int start, int end) {
    if (start >= end) return;
    int pivot = partition(start, end);
    run(start, pivot - 1);
    run(pivot + 1, end);
}

void SortFile::swap(int position1, int position2) {
    double value1 = getValue(position1);
    double value2 = getValue(position2);
    addValue(value1, position2);
    addValue(value2, position1);
}

int SortFile::partition(int start, int end) {
```

```
    int position = start;
    for (int i = start; i < end; i++) {
        if (getValue(i) <= getValue(end)) {
            swap(position++, i);
        }
    }
    swap(position, end);
    return position;
}

double SortFile::getValue(unsigned long position) {
    std::ifstream file;
    double value;
    try {
        file.open(_filename, std::ifstream::binary | std::ifstream::in);
        file.seekg(sizeof (double)*position);
        file.read(reinterpret_cast<char*> (&value), sizeof (double));
        file.close();
    } catch (const std::exception& e) {
        throw "ERROR - can't open the file or get the line ";
    }
    return value;
}

void SortFile::addValue(double value, int position) {
    try {
        std::fstream file(_filename);
        file.seekp(sizeof (double)*position);
        file.write(reinterpret_cast<char*> (&value), sizeof (double));
        file.close();
    } catch (const std::exception& e) {
        throw "ERROR - can't open the file ";
    }
}
```

8.359 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/SorttFile.h File Reference

`#include <string>`

Include dependency graph for SorttFile.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SortFile

8.360 SorttFile.h

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.h to edit this template
 */

/*
 * File:   SorttFile.h
 * Author: rlcancian
 *
 * Created on 20 de novembro de 2019, 22:27
 */

#ifndef SORTTFILE_H
#define SORTTFILE_H

#include <string>

class SortFile {
public:
    SortFile();
    ~SortFile() = default;
public:
    bool sort();
    void setDataFilename(std::string filename);
private:
    void run(int start, int end);
    void swap(int position1, int position2);
    int partition(int start, int end);
    void addValue(double value, int position);
    double getValue(unsigned long position);
private:
    std::string _filename;
    double _numElements = 0;
};

#endif /* SORTTFILE_H */

```

8.361 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/Statistics_if.h File Reference

```
#include <string>
#include "Collector_if.h"
```

Include dependency graph for Statistics_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Statistics_if

8.362 Statistics_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Statistics_if.h
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 13:47
 */

#ifndef STATISTICS_IF_H
#define STATISTICS_IF_H

#include <string>
#include "Collector_if.h"

```

```
class Statistics_if {
public:
    virtual Collector_if* getCollector() = 0;
    virtual void setCollector(Collector_if* collector) = 0;
public:
    virtual unsigned int numElements() = 0;
    virtual double min() = 0;
    virtual double max() = 0;
    virtual double average() = 0;
    virtual double variance() = 0;
    virtual double stddeviation() = 0;
    virtual double variationCoef() = 0;
    virtual double halfWidthConfidenceInterval() = 0;
    virtual unsigned int newSampleSize(double halfWidth) = 0;
    virtual double getConfidenceLevel() = 0;
    virtual void setConfidenceLevel(double confidencelevel) = 0;
};

#endif /* STATISTICS_IF_H */

```

8.363 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDataFile_if.h File Reference

```
#include "CollectorDatafile_if.h"
#include "Statistics_if.h"
```

Include dependency graph for StatisticsDataFile_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class StatisticsDatafile_if

8.364 StatisticsDataFile_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   StatisticsDataFile.h
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Novembro de 2018, 01:16
 */

#ifndef STATISTICSDATAFILE_IF_H
#define STATISTICSDATAFILE_IF_H

#include "CollectorDatafile_if.h"
#include "Statistics_if.h"

class StatisticsDatafile_if : public Statistics_if {
    //public:
    //    virtual CollectorDatafile_if* getCollector() = 0;
    //    virtual void setCollector(Collector_if* collector) = 0;
public:
    virtual double mode() = 0;
    virtual double mediane() = 0;
    virtual double quartil(unsigned short num) = 0;
    virtual double decil(unsigned short num) = 0;
    virtual double centil(unsigned short num) = 0;
    virtual void setHistogramNumClasses(unsigned short num) = 0;
    virtual unsigned short histogramNumClasses() = 0;
    virtual double histogramClassLowerLimit(unsigned short classNum) = 0;
    virtual unsigned int histogramClassFrequency(unsigned short classNum) = 0;

};

#endif /* STATISTICSDATAFILE_IF_H */

```

8.365 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDataFile↩ DefaultImpl.cpp File Reference

```
#include "StatisticsDataFileDefaultImpl.h"
#include "../TraitsKernel.h"
#include <math.h>
```

Include dependency graph for StatisticsDataFileDefaultImpl.cpp:

8.366 StatisticsDataFileDefaultImpl.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
/*
 * File:   StatisticsDataFileDummyImpl.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Novembro de 2018, 01:24
 */

#include "StatisticsDataFileDefaultImpl.h"
#include "../TraitsKernel.h"
#include <math.h>

StatisticsDatafileDefaultImpl1::StatisticsDatafileDefaultImpl1() {
    _collector = new TraitsKernel<StatisticsDatafile_if>::CollectorImplementation();
    _collectorSorted = new TraitsKernel<StatisticsDatafile_if>::CollectorImplementation();
    _z.insert(std::make_pair(0.50, 0));
    _z.insert(std::make_pair(0.80, 1.282));
    _z.insert(std::make_pair(0.90, 1.645));
    _z.insert(std::make_pair(0.95, 1.96));
    _z.insert(std::make_pair(0.975, 2.06));
    _z.insert(std::make_pair(0.98, 2.326));
    _z.insert(std::make_pair(0.99, 2.576));
    _z.insert(std::make_pair(0.995, 2.807));
}

Collector_if* StatisticsDatafileDefaultImpl1::getCollector() {
    return this->_collector;
}

void StatisticsDatafileDefaultImpl1::setCollector(Collector_if* collector) {
    _collector = static_cast<CollectorDatafile_if*> (collector);
}

bool StatisticsDatafileDefaultImpl1::_hasNewValue() {
    if (_numElements < _collector->numElements()) {
        _numElements = _collector->numElements();
        _maxCalculated = false;
        _minCalculated = false;
        _averageCalculated = false;
        _varianceCalculated = false;
        _modeCalculated = false;
        _medianeCalculated = false;
        _stddeviationCalculated = false;
        _variationCoefCalculated = false;
        _halfWidthConfidenceIntervalCalculated = false;
        _newSampleSizeCalculated = false;
        _quartilCalculated = false;
        _decilCalculated = false;
        _centilCalculated = false;
        _histogramNumClassesCalculated = false;
        _histogramClassLowerLimitCalculated = false;
        _histogramClassFrequencyCalculated = false;
        _proportionCalculed = false;
        _fileSorted = false;
        return true;
    } else {
        return false;
    }

}

unsigned int StatisticsDatafileDefaultImpl1::numElements() {
    return _numElements;
}

double StatisticsDatafileDefaultImpl1::min() {
    if (_hasNewValue() || !_minCalculated) {
        for (unsigned long i = 0; i < _collector->numElements(); i++) {
            _min = (_collector->getValue(i) < _min) ? _collector->getValue(i) : _min;
        }
        _minCalculated = true;
    }
    return _min;
}

double StatisticsDatafileDefaultImpl1::max() {
    if (_hasNewValue() || !_maxCalculated) {
        for (unsigned long i = 0; i < _collector->numElements(); i++) {
            _max = (_collector->getValue(i) > _max) ? _collector->getValue(i) : _max;
        }
        _maxCalculated = true;
    }
    return _max;
}

double StatisticsDatafileDefaultImpl1::average() {
    if (_hasNewValue() || !_averageCalculated) {
        valueType sumElements = 0;
```

```
        for (unsigned long i = 0; i < _collector->numElements(); i++) {
            sumElements += _collector->getValue(i);
        }
        _average = sumElements / (valueType) _collector->numElements();
        _averageCalculated = true;
    }
    return _average;
}

double StatisticsDatafileDefaultImpl1::variance() {
    if (_hasNewValue() || !_varianceCalculated) {
        valueType sumElements = 0;
        for (unsigned long i = 0; i < _collector->numElements(); i++) {
            sumElements += pow((_collector->getValue(i) - average()), 2);
        }
        _variance = sumElements / (valueType) (_collector->numElements() - 1);
        _varianceCalculated = true;
    }
    return _variance;
}

double StatisticsDatafileDefaultImpl1::stddeviation() {
    if (_hasNewValue() || !_stddeviationCalculated) {
        _stddeviation = sqrt(variance());
        _stddeviationCalculated = true;
    }
    return _stddeviation;
}

double StatisticsDatafileDefaultImpl1::variationCoef() {
    if (_hasNewValue() || !_variationCoefCalculated) {
        _variationCoef = stddeviation() / average();
        _variationCoefCalculated = true;
    }
    return _variationCoef;
}

double StatisticsDatafileDefaultImpl1::_getNormalProbability(double confidenceLevel) {
    auto search = _z.find(confidenceLevel);
    if (search != _z.end()) {
        return search->second;
    } else {
        return 0.0;
    }
}

double StatisticsDatafileDefaultImpl1::halfWidthConfidenceInterval() {
    if (_hasNewValue() || !_halfWidthConfidenceIntervalCalculated || _confidenceLevel !=
 _lastConfidenceLevel) {
        double z = _getNormalProbability(_confidenceLevel);
        _halfWidthConfidenceInterval = z * stddeviation() / sqrt(_numElements);
        _lastConfidenceLevel = _confidenceLevel;
        _variationCoefCalculated = true;
    }
    return _halfWidthConfidenceInterval;
}

unsigned int StatisticsDatafileDefaultImpl1::newSampleSize(double halfWidth) {
    if (_hasNewValue() || !_newSampleSizeCalculated || _confidenceLevel !=
 _lastNewSampleSizeConfidenceLevel || halfWidth != _lastNewSampleSizeHalfWidth) {
        double z = _getNormalProbability(_confidenceLevel);
        _newSampleSize = pow((z * stddeviation() / halfWidth), 2);
        _lastNewSampleSizeConfidenceLevel = _confidenceLevel;
        _lastNewSampleSizeHalfWidth = halfWidth;
        _newSampleSizeCalculated = true;
    }
    return _newSampleSize;
}

double StatisticsDatafileDefaultImpl1::getConfidenceLevel() {
    return _confidenceLevel;
}

void StatisticsDatafileDefaultImpl1::setConfidenceLevel(double confidencelevel) {
    _confidenceLevel = round(confidencelevel * 100.0) / 100.0;
}

double StatisticsDatafileDefaultImpl1::mode() {
    if (_hasNewValue() || !_modeCalculated) {
        if (!_fileSorted) _sortFile();
        valueType tmpModeValue = _collectorSorted->getValue(0), modeValue;
        unsigned int modeCount = 0, tmpCount = 0;
        for (unsigned long position = 1; position < _collectorSorted->numElements(); position++) {
            valueType value = _collectorSorted->getValue(position);
            if (value == tmpModeValue) {
                tmpCount++;
            } else {
```

```
                if (tmpCount > modeCount) {
                    modeCount = tmpCount;
                    modeValue = tmpModeValue;
                    tmpModeValue = value;
                    tmpCount = 1;
                } else {
                    tmpCount = 1;
                    tmpModeValue = value;
                }
            }
        }
        _mode = modeValue;
        _modeCalculated = true;
    }
    return _mode;
}

double StatisticsDatafileDefaultImpl1::mediane() {
    if (_hasNewValue() || !_medianeCalculated) {
        if (!_fileSorted) _sortFile();

        if (_collectorSorted->numElements() % 2 == 0) {
            valueType tmpValue = _collectorSorted->getValue((_collectorSorted->numElements() / 2) - 1);
            valueType tmpValue2 = _collectorSorted->getValue((_collectorSorted->numElements() / 2));
            _mediane = (tmpValue + tmpValue2) / 2;
        } else {
            _mediane = _collectorSorted->getValue((_collectorSorted->numElements() + 1) / 2);
        }
        _medianeCalculated = true;
    }
    return _mediane;
}

double StatisticsDatafileDefaultImpl1::quartil(unsigned short num) {
    if (_hasNewValue() || !_quartilCalculated || num != _lastQuartilNum) {
        if (num == 2) _quartil = mediane();
        else {
            if (!_fileSorted) _sortFile();

            valueType quartil, tmpValue, tmpValue2;

            unsigned long position = floor(num * _collectorSorted->numElements() / 4) - 1;
            tmpValue = _collectorSorted->getValue(position);
            //tmpValue2 = _collectorSorted->getValue(position + 1);

            _quartil = tmpValue; //(tmpValue + tmpValue2) / 2;
            _quartilCalculated = true;
            _lastQuartilNum = num;
        }
    }
    return _quartil;
}

double StatisticsDatafileDefaultImpl1::decil(unsigned short num) {
    if (_hasNewValue() || !_decilCalculated || num != _lastDecilNum) {
        if (num == 5) _decil = mediane();
        else {
            if (!_fileSorted) _sortFile();
            unsigned long position = ceil(num * _collectorSorted->numElements() / 10);
            _decil = _collectorSorted->getValue(position);
            _decilCalculated = true;
            _lastDecilNum = num;
        }
    }
    return _decil;
}

double StatisticsDatafileDefaultImpl1::centil(unsigned short num) {
    if (_hasNewValue() || !_centilCalculated || num != _lastCentilNum) {
        if (num == 50) _centil = mediane();
        else {
            if (!_fileSorted) _sortFile();
            unsigned long position = ceil(num * _collectorSorted->numElements() / 100);
            _centil = _collectorSorted->getValue(position);
            _centilCalculated = true;
            _lastCentilNum = num;
        }
    }
    return _centil;
}

void StatisticsDatafileDefaultImpl1::setHistogramNumClasses(unsigned short num) {
    _histogramNumClasses = num;
}

unsigned short StatisticsDatafileDefaultImpl1::histogramNumClasses() {
```

```
    if (_hasNewValue() || !_histogramNumClassesCalculated) {
        _histogramNumClasses = ceil(1 + 3.32 * log10(_numElements));
        _histogramNumClassesCalculated = true;
    }
    return _histogramNumClasses;
}

double StatisticsDatafileDefaultImpl1::histogramClassLowerLimit(unsigned short classNum) {
    if (_hasNewValue() || !_histogramClassLowerLimitCalculated || classNum !=
 _LastClassNumHistogramClassLowerLimit) {
        if (classNum == 0)
            _histogramClassLowerLimit = min();
        else
            _histogramClassLowerLimit = min() + classNum * ((max() - min()) / histogramNumClasses());
        _histogramClassLowerLimitCalculated = true;
        _LastClassNumHistogramClassLowerLimit = classNum;
    }
    return _histogramClassLowerLimit;
}

unsigned int StatisticsDatafileDefaultImpl1::histogramClassFrequency(unsigned short classNum) {
    if (_hasNewValue() || !_histogramClassFrequencyCalculated || classNum !=
 _lastClassNumHistogramClassFrequency) {
        if (!_fileSorted) _sortFile();

        valueType classLowerLimit = histogramClassLowerLimit(classNum);
        valueType classUpperLimit = classLowerLimit + ((max() - min()) / histogramNumClasses());
        unsigned int frequency = 0;
        valueType tmpValue;
        for (unsigned long position = 0; position < _collectorSorted->numElements(); position++) {
            tmpValue = _collectorSorted->getValue(position);

            if (tmpValue >= classLowerLimit && tmpValue < classUpperLimit) {
                frequency++;
            } else if (tmpValue >= classUpperLimit) {
                break;
            }
        }
        _histogramClassFrequency = frequency;
        _histogramClassFrequencyCalculated = true;
        _lastClassNumHistogramClassFrequency = classNum;
    }
    return _histogramClassFrequency;
}

void StatisticsDatafileDefaultImpl1::_sortFile() {

    if (!_fileSortedCreated) {
        _collectorSorted->setDataFilename(_collector->getDataFilename() + "_sorted");
        _collectorSorted->clear();
        _fileSortedCreated = true;
    }

    if (_collectorSorted->numElements() < _collector->numElements()) {
        for (unsigned long position = _collectorSorted->numElements(); position <
 _collector->numElements(); position++) {
            _collectorSorted->addValue(_collector->getValue(position));
        }
    }
    sort->setDataFilename(_collectorSorted->getDataFilename());
    sort->sort();
    _fileSorted = true;
}
```

8.367 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDataFileDefaultImpl.h File Reference

```
#include "StatisticsDataFile_if.h"
#include "SorttFile.h"
#include <limits.h>
#include <map>
```

Include dependency graph for StatisticsDataFileDefaultImpl.h:

This graph shows which files directly or indirectly include this file:

Classes

- class StatisticsDatafileDefaultImpl1

Typedefs

- typedef double valueType

8.367.1 Typedef Documentation

8.367.1.1 valueType `typedef double valueType`

Definition at line 22 of file StatisticsDataFileDefaultImpl.h.

8.368 StatisticsDataFileDefaultImpl.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   StatisticsDataFileDefaultImpl.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 22 de Novembro de 2018, 01:24
00012  */
00013
00014 #ifndef STATISTICSDATAFILEDEFAULTIMPL_H
00015 #define STATISTICSDATAFILEDEFAULTIMPL_H
00016
00017 #include "StatisticsDataFile_if.h"
00018 #include "SorttFile.h"
00019 #include <limits.h>
00020 #include <map>
00021
00022 typedef double valueType;
00023
00024 class StatisticsDatafileDefaultImpl1 : public StatisticsDatafile_if {
00025 public:
00026     StatisticsDatafileDefaultImpl1();
00027     virtual ~StatisticsDatafileDefaultImpl1() = default;
00028 public:
00029     virtual Collector_if* getCollector();
00030     virtual void setCollector(Collector_if* collector);
00031 public:
00032     virtual unsigned int numElements();
00033     virtual double min();
00034     virtual double max();
00035     virtual double average();
00036     virtual double variance();
00037     virtual double stddeviation();
00038     virtual double variationCoef();
00039     virtual double halfWidthConfidenceInterval();
00040     virtual unsigned int newSampleSize(double halfWidth);
00041     virtual double getConfidenceLevel();
00042     virtual void setConfidenceLevel(double confidencelevel);
00043 public:
00044     virtual double mode();
00045     virtual double mediane();
00046     virtual double quartil(unsigned short num);
00047     virtual double decil(unsigned short num);
00048     virtual double centil(unsigned short num);
00049     virtual void setHistogramNumClasses(unsigned short num);
00050     virtual unsigned short histogramNumClasses();
00051     virtual double histogramClassLowerLimit(unsigned short classNum);
00052     virtual unsigned int histogramClassFrequency(unsigned short classNum);
00053 private:
00054     void _sortFile();
00055     bool _hasNewValue();
00056     double _getNormalProbability(double confidenceLevel);
00057 private:
00058     CollectorDatafile_if* _collector;
00059     CollectorDatafile_if* _collectorSorted;
00060     //CollectorDatafile_if* _collector;
00061     SortFile * sort = new SortFile();
00062     double _confidenceLevel = 0.95;
00063     unsigned long _numElements = 0;
00064     std::map<double, double> _z;
00065     bool _fileSorted = false;
00066     bool _fileSortedCreated = false;
00067     //unsigned long _numElements = 0;
00068     bool _numElementsCalculated = false;
00069     double _max = INT_MIN;
00070     bool _maxCalculated = false;
00071     double _min = INT_MAX;
00072     bool _minCalculated = false;
00073     bool _modeCalculated = false;
00074     double _mode = 0;
00075     bool _medianeCalculated = false;
00076     double _mediane = 0;
00077     double _average = 0;
00078     bool _averageCalculated = false;
00079     double _variance = 0;
00080     bool _varianceCalculated = false;
00081     double _sumSquareElements = 0;
00082     double _sumElements = 0;
00083     double _stddeviation = 0;
00084     bool _stddeviationCalculated = false;
00085     double _variationCoef = 0;
```

```
    bool _variationCoefCalculated = false;
    bool _halfWidthConfidenceIntervalCalculated = false;
    double _lastConfidenceLevel = 0;
    double _halfWidthConfidenceInterval = 0;
    bool _newSampleSizeCalculated = false;
    double _lastNewSampleSizeConfidenceLevel = 0;
    double _lastNewSampleSizeHalfWidth = 0;
    double _newSampleSize = 0;
    bool _quartilCalculated = false;
    unsigned short _lastQuartilNum = 0;
    double _quartil = 0;
    bool _decilCalculated = false;
    unsigned short _lastDecilNum = 0;
    double _decil = 0;
    bool _centilCalculated = false;
    unsigned short _lastCentilNum = 0;
    double _centil = 0;
    bool _histogramNumClassesCalculated = false;
    unsigned short _histogramNumClasses = 1;
    bool _histogramClassLowerLimitCalculated = false;
    unsigned short _LastClassNumHistogramClassLowerLimit = 0;
    double _histogramClassLowerLimit = 0;
    bool _histogramClassFrequencyCalculated = false;
    unsigned short _lastClassNumHistogramClassFrequency = 0;
    unsigned int _histogramClassFrequency = 0;
    bool _proportionCalculed = false;
    double _proportion = 0;
};

#endif /* STATISTICSDATAFILEDEFAULTIMPL_H */

```

8.369 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDefault↩ Impl1.cpp File Reference

```
#include <complex>
#include "StatisticsDefaultImpl1.h"
#include "../TraitsKernel.h"
```

Include dependency graph for StatisticsDefaultImpl1.cpp:

8.370 StatisticsDefaultImpl1.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   StatisticsDefaultImpl1.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 1 de Agosto de 2018, 21:03
 */

#include <complex>

#include "StatisticsDefaultImpl1.h"
#include "../TraitsKernel.h"
//#include "Integrator_if.h"
//#include "ProbDistribDefaultImpl1.h"

//using namespace GenesysKernel;
```

```

StatisticsDefaultImpl1::StatisticsDefaultImpl1() {
    //_collector = new TraitsKernel<Statistics_if>::CollectorImplementation();
    _collector = new TraitsKernel<ModelComponent>::StatisticsCollector_CollectorImplementation();
    _collector->setAddValueHandler(setCollectorAddValueHandler(&StatisticsDefaultImpl1::collectorAddHandler, this));
    _collector->setClearHandler(setCollectorClearHandler(&StatisticsDefaultImpl1::collectorClearHandler, this));
    //_collector->setAddValueHandler(std::bind(&StatisticsDefaultImpl1::collectorAddHandler, this, std::placeholders::_1));
    this->initStatistics();
}

StatisticsDefaultImpl1::StatisticsDefaultImpl1(Collector_if* collector) {
    _collector = collector;
    _collector->setAddValueHandler(setCollectorAddValueHandler(&StatisticsDefaultImpl1::collectorAddHandler, this));
    _collector->setClearHandler(setCollectorClearHandler(&StatisticsDefaultImpl1::collectorClearHandler, this));
    //_collector->setAddValueHandler(std::bind(&StatisticsDefaultImpl1::collectorAddHandler, this, std::placeholders::_1));
    this->initStatistics();
}

void StatisticsDefaultImpl1::collectorAddHandler(double newValue) {
    _elems = _collector->numElements();
    if (newValue < _min) {
        _min = newValue;
    }
    if (newValue > _max) {
        _max = newValue;
    }
    // alternative 1
    _sum += newValue;
    _sumSquare += newValue*newValue;
    _average = _sum / _elems;
    _variance = _sumSquare / _elems - _average*_average;
    // alternative 2
    //_average = (_average * (elems - 1) + newValue) / elems;  // this approach propagates the numeric error
    //_variance = (_variance * (elems - 1) + pow(newValue - _average, 2)) / elems;  // this approach propagates the numeric error
    _stddeviation = sqrt(_variance);
    _variationCoef = (_average != 0 ? _stddeviation / _average : 0.0);
    _halfWidth = _criticalTn_1 * (_stddeviation / std::sqrt(_elems));
}

void StatisticsDefaultImpl1::collectorClearHandler() {
    this->initStatistics();
}

void StatisticsDefaultImpl1::initStatistics() {
    _elems = 0;
    _min = +1e+99;
    _max = -1e+99;
    _sum = 0.0;
    _sumSquare = 0.0;
    _average = 0.0;
    _variance = 0.0;
    _stddeviation = 0.0;
    _variationCoef = 0.0;
    _halfWidth = 0.0;
}

unsigned int StatisticsDefaultImpl1::numElements() {
    return this->getCollector()->numElements();
}

double StatisticsDefaultImpl1::min() {
    if (_elems > 0)
        return _min;
    else
        return 0.0;
}

double StatisticsDefaultImpl1::max() {
    if (_elems > 0)
        return _max;
    else
        return 0.0;
}

double StatisticsDefaultImpl1::average() {
```

```
    return _average;
}

double StatisticsDefaultImpl1::variance() {
    return _variance;
}

double StatisticsDefaultImpl1::stddeviation() {
    return _stddeviation;
}

double StatisticsDefaultImpl1::variationCoef() {
    return _variationCoef;
}

double StatisticsDefaultImpl1::halfWidthConfidenceInterval() {
    return _halfWidth;
}

void StatisticsDefaultImpl1::setConfidenceLevel(double confidencelevel) {
    _confidenceLevel = confidencelevel;
    //Integrator_if* integrator = new TraitsKernel<Integrator_if>::Implementation();
    _criticalTn_1 = 1.96; //integrator->integrate()

}

double StatisticsDefaultImpl1::getConfidenceLevel() {
    return _confidenceLevel;
}

unsigned int StatisticsDefaultImpl1::newSampleSize(double halfWidth) {
    return 0;
}

Collector_if* StatisticsDefaultImpl1::getCollector() {
    return this->_collector;
}

void StatisticsDefaultImpl1::setCollector(Collector_if* collector) {
    this->_collector = collector;
}
```

8.371 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/statistics/StatisticsDefault↩ Impl1.h File Reference

```
#include "Statistics_if.h"
#include "Collector_if.h"
```

Include dependency graph for StatisticsDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class StatisticsDefaultImpl1

8.372 StatisticsDefaultImpl1.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   StatisticsDefaultImpl1.h
 * Author: rafael.luiz.cancian
 *
 * Created on 1 de Agosto de 2018, 21:03
 */

#ifndef STATISTICSDEFAULTIMPL1_H
#define STATISTICSDEFAULTIMPL1_H

#include "Statistics_if.h"
#include "Collector_if.h"
//namespace GenesysKernel {

class StatisticsDefaultImpl1 : public Statistics_if {
public:
    StatisticsDefaultImpl1();
    StatisticsDefaultImpl1(Collector_if* collector);
    virtual ~StatisticsDefaultImpl1() = default;
```

```
public:
    virtual Collector_if* getCollector();
    virtual void setCollector(Collector_if* collector);
public:
    virtual unsigned int numElements();
    virtual double min();
    virtual double max();
    virtual double average();
    virtual double variance();
    virtual double stddeviation();
    virtual double variationCoef();
    virtual double halfWidthConfidenceInterval();
    virtual unsigned int newSampleSize(double halfWidth);
    virtual double getConfidenceLevel();
    virtual void setConfidenceLevel(double confidencelevel);
private:
    void collectorAddHandler(double newValue);
    void collectorClearHandler();
    void initStatistics();
private:
    Collector_if* _collector;
    unsigned long _elems;
    double _sum;
    double _sumSquare;
    double _min;
    double _max;
    double _average;
    double _variance;
    double _stddeviation;
    double _variationCoef;
    double _confidenceLevel = 0.95;
    double _criticalTn_1 = 1.96;
    double _halfWidth;
};
//namespace\\}
#endif /* STATISTICSDEFAULTIMPL1_H */
```

8.373 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/TraitsKernel.h File Reference

```
#include "simulator/ModelPersistenceDefaultImpl1.h"
#include "simulator/SimulationReporterDefaultImpl1.h"
#include "simulator/Counter.h"
#include "simulator/ModelCheckerDefaultImpl1.h"
#include "simulator/PluginConnectorDummyImpl1.h"
#include "simulator/ParserDefaultImpl2.h"
#include "statistics/CollectorDatafileDefaultImpl1.h"
#include "statistics/CollectorDefaultImpl1.h"
#include "statistics/StatisticsDefaultImpl1.h"
#include "statistics/StatisticsDataFileDefaultImpl.h"
#include "statistics/SamplerDefaultImpl1.h"
```

Include dependency graph for TraitsKernel.h:

This graph shows which files directly or indirectly include this file:

Classes

- struct TraitsKernel< T >
- struct TraitsKernel< SimulationReporter_if >
- struct TraitsKernel< PluginConnector_if >
- struct TraitsKernel< Parser_if >
- struct TraitsKernel< Model >
- struct TraitsKernel< ModelComponent >
- struct TraitsKernel< ModelElement >
- struct TraitsKernel< ModelChecker_if >
- struct TraitsKernel< ModelPersistence_if >
- struct TraitsKernel< Statistics_if >
- struct TraitsKernel< StatisticsDatafile_if >
- struct TraitsKernel< Sampler_if >
- struct TraitsKernel< Collector_if >

8.374 TraitsKernel.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Traits.h
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 19:36
 */


#ifndef TRAITSKERNEL_H
#define TRAITSKERNEL_H


#include "simulator/ModelPersistenceDefaultImpl1.h"
#include "simulator/SimulationReporterDefaultImpl1.h"
#include "simulator/Counter.h"
#include "simulator/ModelCheckerDefaultImpl1.h"
#include "simulator/PluginConnectorDummyImpl1.h"
#include "simulator/ParserDefaultImpl2.h"

#include "statistics/CollectorDatafileDefaultImpl1.h"
#include "statistics/CollectorDefaultImpl1.h"
#include "statistics/StatisticsDefaultImpl1.h"
#include "statistics/StatisticsDataFileDefaultImpl.h"
#include "statistics/SamplerDefaultImpl1.h"



//namespace GenesysKernel {

template <typename T>
struct TraitsKernel {
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results;
};

/*
 *  Simulation and Simulation Parts
 */

template <> struct TraitsKernel<SimulationReporter_if> {
    typedef SimulationReporterDefaultImpl1 Implementation;
    typedef Counter CounterImplementation;
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results;
};

template <> struct TraitsKernel<PluginConnector_if> {
    typedef PluginConnectorDummyImpl1 Implementation;
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L4_warning;
};

template <> struct TraitsKernel<Parser_if> {
    typedef ParserDefaultImpl2 Implementation;
};
```

```

/*
 *  Model and Model Parts
 */

template <> struct TraitsKernel<Model> {
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L5_event;
};

template <> struct TraitsKernel<ModelComponent> {
    typedef StatisticsDefaultImpl1 StatisticsCollector_StatisticsImplementation;
    typedef CollectorDefaultImpl1 StatisticsCollector_CollectorImplementation;
    static constexpr bool reportStatistics = true;
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results;
};

template <> struct TraitsKernel<ModelElement> {
    static constexpr bool reportStatistics = true;
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results;
};

template <> struct TraitsKernel<ModelChecker_if> {
    typedef ModelCheckerDefaultImpl1 Implementation;
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results;
};

template <> struct TraitsKernel<ModelPersistence_if> {
    typedef ModelPersistenceDefaultImpl1 Implementation;
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L2_results;
};

/*
 *  Statistics
 */
template <> struct TraitsKernel<Statistics_if> {
    typedef StatisticsDefaultImpl1 Implementation;
    typedef CollectorDefaultImpl1 CollectorImplementation;
    typedef double DataType; // TODO: not used yet. Change all classes that collect statistics to this type (so classes that deal with erros and bit limits can be assigned to it
    static constexpr double SignificanceLevel = 0.05;
};

template <> struct TraitsKernel<StatisticsDatafile_if> {
    typedef StatisticsDatafileDefaultImpl1 Implementation;
    typedef CollectorDatafileDefaultImpl1 CollectorImplementation;
    static constexpr double SignificanceLevel = 0.05;
};

template <> struct TraitsKernel<Sampler_if> {
    typedef SamplerDefaultImpl1 Implementation;
    typedef SamplerDefaultImpl1::DefaultImpl1RNG_Parameters Parameters;
};

template <> struct TraitsKernel<Collector_if> {
    typedef CollectorDatafileDefaultImpl1 Implementation;
    typedef double DataType; // TODO: not used yet. Change all classes that collect statistics to this type (so classes that deal with erros and bit limits can be assigned to it
};
//namespace\\}

#endif /* TRAITSKERNEL_H */

```

8.375 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/util/Exact.h File Reference

```
#include <iostream>
#include <iomanip>
```

Include dependency graph for Exact.h:

Classes

- class Exact

8.376 Exact.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Exact.h
 * Author: rlcancian
 *
 * Created on 12 de agosto de 2021, 18:55
 */

#ifndef Exact_H
#define Exact_H

#include <iostream>
#include <iomanip>

class Exact {
    //friend ostream &operator«(ostream&, const Exact &);
    //friend istream &operator»(istream&, Exact &);
public:
    Exact() {
    };
    Exact(int num, int den = 1) {
        _num = num;
        _den = den;
    };
    const Exact &operator+(const Exact &right) {
        Exact e;
        e._num = this->_num * right._den + right._num * this->_den;
        e._den = this->_den * right._den;
        return &e;
    }
    const Exact &operator+=(const Exact &right) {
        this->_num = this->_num * right._den + right._num * this->_den;
        this->_den *= right._den;
    }
    const Exact &operator-=(const Exact &right) {
        this->_num = this->_num * right._den - right._num * this->_den;
        this->_den *= right._den;
    }
    const Exact &operator*=(const Exact &right) {
        this->_num *= right._num;
```

```
        this->_den *= right._den;
    }
    const Exact &operator/=(const Exact &right) {
        this->_num *= right._num;
        this->_den *= right._den;
    }
    const Exact &operator=(const Exact &right) {
        this->_num = right._num;
        this->_den = right._den;
    }
    const Exact &operator=(const int &right) {
        this->_num = right;
        this->_den = 1;
    }
    bool operator==(const Exact &right) {
        // verify proportionality is missing
        return (this->_num == right._num) && (this->_den == right._den);
    }
    bool operator==(const double &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) == right;
    }
    bool operator<(const Exact &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) < (right._num / right._den);
    }
    bool operator<(const double &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) < right;
    }
    bool operator<=(const Exact &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) <= (right._num / right._den);
    }
    bool operator<=(const double &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) <= right;
    }
    bool operator>(const Exact &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) > (right._num / right._den);
    }
    bool operator>(const double &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) > right;
    }
    bool operator>=(const Exact &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) >= (right._num / right._den);
    }
    bool operator>=(const double &right) {
        // verify proportionality is missing
        return (this->_num / this->_den) >= right;
    }
    void Simplify() {
    }
    double toFloat() {
        return _num / _den;
    };
protected:
private:
    double _num, _den;
};


/*
ostream &operator«(ostream &output, const Exact &num) {
    output « num._num « "/" « num._den;
    return output;
}

/*
istream &operator»(istream &input, Exact &num) {
    input » num._num;
    input » num._dem;
    return input;
}
 */


#endif /* Exact_H */

```

8.377 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/util/List.h File Reference

```
#include <string>
#include <list>
#include <map>
#include <iterator>
#include <functional>
#include <algorithm>
#include "Util.h"
#include "../simulator/ModelElement.h"
```

Include dependency graph for List.h:

This graph shows which files directly or indirectly include this file:

Classes

- class List< T >

8.378 List.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   TManager.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 12:55
00012  */
00013
00014 #ifndef LISTMANAGER_H
00015 #define LISTMANAGER_H
00016
00017 #include <string>
00018 #include <list>
00019 #include <map>
```

```
#include <iterator>
#include <functional>
#include <algorithm>
#include "Util.h"
#include "../simulator/ModelElement.h"

//class Simulator;

template <typename T>
class List {
public:
    using CompFunct = std::function<bool(const T, const T) >;
public:
    List();
    virtual ~List() = default;
public: // direct access to list
    unsigned int size();
    bool empty();
    void clear();
    void pop_front();
    template<class Compare>
    void sort(Compare comp);
    std::list<T>* list() const;
public: // new methods
    T create();
    template<typename U>
    T create(U arg);
    std::string show();
    typename std::list<T>::iterator find(T element);
    //int rankOf(T element); ///< returns the position (1st position=0) of the element if found, or negative value if not found
public: // improved (easier) methods
    void insert(T element);
    void remove(T element);
    void setAtRank(unsigned int rank, T element);
    T getAtRank(unsigned int rank);
    T next();
    T front();
    T last();
    T previous();
    T current(); // get current element on the list (the last used)
    void setSortFunc(CompFunct _sortFunc);
private:
    //std::map<Util::identitifcation, T>* _map;
    std::list<T>* _list;
    CompFunct _sortFunc{[](const T, const T) {
            return false;
        }};
    typename std::list<T>::iterator _it;
};
template <typename T>
List<T>::List() {
    //_map = new std::map<Util::identitifcation, T>();
    _list = new std::list<T>();
    _it = _list->begin();
}
template <typename T>
std::list<T>* List<T>::list() const {
    return _list;
}
template <typename T>
unsigned int List<T>::size() {
    return _list->size();
}

//template <typename T>
//List<T>::List(const List& orig) {
//}

//template <typename T>
//List<T>::~List() {
//}
template <typename T>
std::string List<T>::show() {
    int i = 0;
    std::string text = "{";
    for (typename std::list<T>::iterator it = _list->begin(); it != _list->end(); it++, i++) {
        text += "[" + std::to_string(i) + "]=(" + (*it)->show() + ),";
    }
    text += "}";
    return text;
}
template <typename T>
void List<T>::insert(T element) {
    _list->insert(std::upper_bound(_list->begin(), _list->end(), element, _sortFunc), element);
}
template <typename T>
```

```
bool List<T>::empty() {
    return _list->empty();
}
template <typename T>
void List<T>::pop_front() {
    typename std::list<T>::iterator itTemp = _list->begin();
    _list->pop_front();
    if (_it == itTemp) { /*  \todo: +: check this */
        _it = _list->begin(); // if it points to the removed element, then changes to begin
    }
}
template <typename T>
void List<T>::remove(T element) {
    _list->remove(element);
    if ((*_it) == element) { /*  \todo: +: check this */
        _it = _list->begin(); // if it points to the removed element, then changes to begin
    }
}
template <typename T>
T List<T>::create() {
    return new T();
}
template <typename T>
void List<T>::clear() {
    _list->clear();
}
template <typename T>
T List<T>::getAtRank(unsigned int rank) {
    unsigned int thisRank = 0;
    for (typename std::list<T>::iterator it = _list->begin(); it != _list->end(); it++) {
        if (rank == thisRank) {
            return (*it);
        } else {
            thisRank++;
        }
    }
    return 0; /* \todo: Invalid return depends on T. If T is pointer, nullptr works fine. If T is double, it does not. I just let (*it), buut it is not nice*/
}
template <typename T>
void List<T>::setAtRank(unsigned int rank, T element) {
    if (rank == _list->size()) {
        _list->insert(_list->end(), element);
    } else {
        unsigned int thisRank = 0;
        for (typename std::list<T>::iterator it = _list->begin(); it != _list->end(); it++) {
            if (rank == thisRank) {
                *it = element;
                return;
            } else {
                thisRank++;
            }
        }
    }
}
template <typename T>
T List<T>::next() {
    _it++;
    if (_it != _list->end())
        return (*_it);
    else
        return nullptr;

}
template <typename T>
typename std::list<T>::iterator List<T>::find(T element) {
    for (typename std::list<T>::iterator it = _list->begin(); it != _list->end(); it++) {
        if ((*it) == element) {
            return it;
        }
    }
    return _list->end(); /*  \todo:+-: check nullptr or invalid iterator when not found */
    //return nullptr;
}
/*
template <typename T>
int List<T>::rankOf(T element) {
    int rank = 0;
    for (typename std::list<T>::iterator it = _list->begin(); it != _list->end(); it++) {
    if ((*it) == element) {
        return rank;
    } else
        rank++;
    }
    return -1; // not found -> negative rank
}
 */
```

```

template <typename T>
T List<T>::front() {
    _it = _list->begin();
    //if (_it != _list->end())
    return (*_it);
    //else
    //return dynamic_cast<T>(nullptr);
}
template <typename T>
T List<T>::last() {
    _it = _list->end();
    _it--;
    //if (_it != _list->end()) // \todo: CHECK!!!
    return (*_it);
    //else return nullptr;
}
template <typename T>
T List<T>::previous() {
    _it--; // \todo: CHECK!!!
    return (*_it);
}
template <typename T>
T List<T>::current() {
    /* \todo: To implement (i thing it's just to check). Must actualize _it on other methods when other elements are accessed */
    return (*_it);
}
template <typename T>
void List<T>::setSortFunc(CompFunct _sortFunc) {
    this->_sortFunc = _sortFunc;
}
template <typename T>
template<typename U>
T List<T>::create(U arg) {
    return T(arg);
}
template <typename T>
template<class Compare>
void List<T>::sort(Compare comp) {
    _list->sort(comp);
}

#endif /* LISTMANAGER_H */

```

8.379 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/util/Util.cpp File Reference

```
#include "Util.h"
#include <typeinfo>
#include <map>
```

Include dependency graph for Util.cpp:

8.380 Util.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Util.cpp
```

```
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 13:02
 */

#include "Util.h"
#include <typeinfo>
#include <map>

//using namespace GenesysKernel;

Util::identification Util::_S_lastId = 0;
unsigned int Util::_S_indentation;
std::map<std::string, Util::identification> Util::_S_lastIdOfType = std::map<std::string, Util::identification>();
std::map<std::string, std::string> Util::_S_TypeOf = std::map<std::string, std::string>();

void Util::IncIndent() {
    Util::_S_indentation++;
}

void Util::SetIndent(const unsigned short indent) {
    Util::_S_indentation = indent;
}

void Util::DecIndent() {
    Util::_S_indentation--;
}

void Util::SepKeyVal(std::string str, std::string *key, std::string *value) {
    // \todo: Check pointers when spliting string. There is an error
    //char *c;
    bool settingKey = true;
    //key = new std::string();
    //value = new std::string();
    key->clear();
    value->clear();
    for (std::string::iterator it = str.begin(); it != str.end(); it++) {
        if (settingKey) {
            if ((*it) != '=') {
                key->append(new char((*it)));
            } else {
                settingKey = false;
            }
        } else {
            value->append(new char((*it)));
        }
    }
}

std::string Util::Indent() {
    std::string spaces = "";
    for (unsigned int i = 0; i < Util::_S_indentation; i++) {
        spaces += "|   ";
    }
    return spaces;
}

std::string Util::SetW(std::string text, unsigned short width) {
    std::string spaces(width, ' ');
    std::string result = text + spaces;
    return result.substr(0, width);
}

std::string Util::StrTimeUnitShort(Util::TimeUnit timeUnit) {
    switch (static_cast<int> (timeUnit)) {
        case 1: return "ps";
        case 2: return "ns";
        case 3: return "us";
        case 4: return "ms";
        case 5: return "s";
        case 6: return "min";
        case 7: return "h";
        case 8: return "d";
        case 9: return "w";
        default: break;
    }
    return "";
}

std::string Util::StrTimeUnitLong(Util::TimeUnit timeUnit) {
    switch (static_cast<int> (timeUnit)) {
        case 1: return "picosecond";
        case 2: return "nanosecond";
        case 3: return "microsecond";
        case 4: return "milisecond";
        case 5: return "second";
```

```
        case 6: return "minute";
        case 7: return "hour";
        case 8: return "day";
        case 9: return "week";
    }
    return "";
}

Util::identification Util::GenerateNewId() {
    Util::_S_lastId++;
    return Util::_S_lastId;
}

Util::identification Util::GenerateNewIdOfType(std::string objtype) {
    std::map<std::string, Util::identification>::iterator it = Util::_S_lastIdOfType.find(objtype);
    if (it == Util::_S_lastIdOfType.end()) {
        // a new one. create the pair
        Util::_S_lastIdOfType.insert(std::pair<std::string, Util::identification>(objtype, 0));
        it = Util::_S_lastIdOfType.find(objtype);
    }
    Util::identification next = (*it).second;
    next++;
    (*it).second = next;
    return (*it).second;
}

Util::identification Util::GetLastIdOfType(std::string objtype) {
    std::map<std::string, Util::identification>::iterator it = Util::_S_lastIdOfType.find(objtype);
    if (it == Util::_S_lastIdOfType.end()) {
        // a new one. create the pair
        Util::_S_lastIdOfType.insert(std::pair<std::string, Util::identification>(objtype, 0));
        it = Util::_S_lastIdOfType.find(objtype);
    }
    //Util::identification next = (*it).second;
    //next++;
    //(*it).second = next;
    return (*it).second;

}

void Util::ResetIdOfType(std::string objtype) {
    std::map<std::string, Util::identification>::iterator it = Util::_S_lastIdOfType.find(objtype);
    if (it == Util::_S_lastIdOfType.end()) {
        // a new one. create the pair
        Util::_S_lastIdOfType.insert(std::pair<std::string, Util::identification>(objtype, 0));
        it = Util::_S_lastIdOfType.find(objtype);
    }
    (*it).second = 0;
}

double Util::TimeUnitConvert(Util::TimeUnit timeUnit1, Util::TimeUnit timeUnit2) {
    double scaleValues[] = {1.0, 1000.0, 1000.0, 1000.0, 1000.0, 1000.0, 60.0, 60.0, 24.0, 7.0};
    // picosecond = 1, nanosecond = 2, microsecond = 3, milisecond = 4, second = 5, minute = 6, hour =
    7, day = 8, week = 9
    double res = 1.0;
    int intTimeUnit1, intTimeUnit2;
    intTimeUnit1 = static_cast<int> (timeUnit1);
    intTimeUnit2 = static_cast<int> (timeUnit2);
    if (intTimeUnit1 <= intTimeUnit2) {
        for (int i = intTimeUnit1 + 1; i <= intTimeUnit2; i++) {
            res /= scaleValues[i];
        }
    } else {
        for (int i = intTimeUnit2 + 1; i <= intTimeUnit1; i++) {
            res *= scaleValues[i];
        }
    }
    return res;
}
```

8.381 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/kernel/util/Util.h File Reference

```
#include <map>
#include <typeinfo>
#include <string>
#include <cctype>
#include <algorithm>
#include <locale>
```

```
#include <stdio.h>
#include <limits.h>
#include <unistd.h>
#include <vector>
#include <dirent.h>
#include <errno.h>
#include <iostream>
#include <sys/stat.h>
```

Include dependency graph for Util.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Util

Functions

- static std::string map2str (std::map< std::string, std::string > *mapss)
- static std::string strTruncIfInt (std::string strValue)
- static std::string trim (std::string str)
- static void trimwithin (std::string &str)
- static std::string getFileName (const std::string &s)
- static std::string getPath (const std::string &s)
- static std::string LoadField (std::map< std::string, std::string > *fields, std::string fieldName, std::string defaultValue="")
- static double LoadField (std::map< std::string, std::string > *fields, std::string fieldName, double defaultValue)
- static unsigned int LoadField (std::map< std::string, std::string > *fields, std::string fieldName, unsigned int defaultValue)
- static int LoadField (std::map< std::string, std::string > *fields, std::string fieldName, int defaultValue)
- static Util::TimeUnit LoadField (std::map< std::string, std::string > *fields, std::string fieldName, Util::TimeUnit defaultValue)
- static void SaveField (std::map< std::string, std::string > *fields, std::string fieldName, std::string fieldValue, const std::string fieldDefaultValue="", const bool saveIfEqualsDefault=false)

- static void SaveField (std::map< std::string, std::string > *fields, std::string fieldName, double fieldValue, const double fieldDefaultValue)
- static void SaveField (std::map< std::string, std::string > *fields, std::string fieldName, unsigned int field↩ Value, const unsigned int fieldDefaultValue)
- static void SaveField (std::map< std::string, std::string > *fields, std::string fieldName, unsigned int field↩ Value)
- static void SaveField (std::map< std::string, std::string > *fields, std::string fieldName, int fieldValue, const int fieldDefaultValue)
- static void SaveField (std::map< std::string, std::string > *fields, std::string fieldName, Util::TimeUnit field↩ Value, const Util::TimeUnit fieldDefaultValue, const bool saveDefaultValue=false)

8.381.1 Function Documentation

8.381.1.1 getFileName()

```
static std::string getFileName (
    const std::string & s ) [inline], [static]
```

Definition at line 86 of file Util.h.

Here is the caller graph for this function:

8.381.1.2 getPath()

```
static std::string getPath (
    const std::string & s ) [inline], [static]
```

Definition at line 94 of file Util.h.

8.381.1.3 LoadField() [1/5]

```
static double LoadField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    double defaultValue ) [inline], [static]
```

Definition at line 255 of file Util.h.

8.381.1.4 LoadField() [2/5]

```
static int LoadField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    int defaultValue ) [inline], [static]
```

Definition at line 261 of file Util.h.

8.381.1.5 LoadField() [3/5]

```
static std::string LoadField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    std::string defaultValue = "" ) [inline], [static]
```

Definition at line 252 of file Util.h.

8.381.1.6 LoadField() [4/5]

```
static unsigned int LoadField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    unsigned int defaultValue ) [inline], [static]
```

Definition at line 258 of file Util.h.

8.381.1.7 LoadField() [5/5]

```
static Util::TimeUnit LoadField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    Util::TimeUnit defaultValue ) [inline], [static]
```

Definition at line 264 of file Util.h.

8.381.1.8 map2str()

```
static std::string map2str (
    std::map< std::string, std::string > * mapss ) [inline], [static]
```

Definition at line 36 of file Util.h.

8.381.1.9 SaveField() [1/6]

```
static void SaveField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    double fieldValue,
    const double fieldDefaultValue ) [inline], [static]
```

Definition at line 273 of file Util.h.

8.381.1.10 SaveField() [2/6]

```
static void SaveField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    int fieldValue,
    const int fieldDefaultValue ) [inline], [static]
```

Definition at line 284 of file Util.h.

8.381.1.11 SaveField() [3/6]

```
static void SaveField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    std::string fieldValue,
    const std::string fieldDefaultValue = "",
    const bool saveIfEqualsDefault = false ) [inline], [static]
```

Definition at line 269 of file Util.h.

8.381.1.12 SaveField() [4/6]

```
static void SaveField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    unsigned int fieldValue ) [inline], [static]
```

Definition at line 281 of file Util.h.

8.381.1.13 SaveField() [5/6]

```
static void SaveField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    unsigned int fieldValue,
    const unsigned int fieldDefaultValue ) [inline], [static]
```

Definition at line 277 of file Util.h.

8.381.1.14 SaveField() [6/6]

```
static void SaveField (
    std::map< std::string, std::string > * fields,
    std::string fieldName,
    Util::TimeUnit fieldValue,
    const Util::TimeUnit fieldDefaultValue,
    const bool saveDefaultValue = false ) [inline], [static]
```

Definition at line 288 of file Util.h.

8.381.1.15 strTruncIfInt()

```
static std::string strTruncIfInt (
    std::string strValue ) [inline], [static]
```

Definition at line 67 of file Util.h.

Here is the caller graph for this function:

8.381.1.16 trim()

```
static std::string trim (
    std::string str ) [inline], [static]
```

Definition at line 73 of file Util.h.

Here is the caller graph for this function:

8.381.1.17 trimwithin() static void trimwithin (std::string & *str*) [inline], [static]

Definition at line 80 of file Util.h.

Here is the caller graph for this function:

8.382 Util.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Util.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 13:02
00012  */
00013
00014 #ifndef UTIL_H
00015 #define UTIL_H
00016
00017 #include <map>
00018 #include <typeinfo>
00019 #include <string>
00020 #include <cctype>
00021 #include <algorithm>
00022 #include <cctype>
00023 #include <locale>
00024 #include <stdio.h>
00025 // dir
00026 #include <limits.h>
00027 #include <unistd.h>
00028 #include <vector>
00029 #include <dirent.h>
00030 #include <errno.h>
00031 #include <iostream>
00032 #include <sys/stat.h>
00033
00034 //namespace GenesysKernel {
00035 // trim from start (in place)
00036 static inline std::string map2str(std::map<std::string, std::string>* mapss) {
00037     std::string res = "";
00038     for (std::map<std::string, std::string>::iterator it = mapss->begin(); it != mapss->end(); it++) {
00039         res += (*it).first + "=" + (*it).second + " ";
00040     }
00041     res = res.substr(0, res.length() - 1);
00042     return res;
00043 }
00044 /*
00045 static inline void ltrim(std::string &s) {
00046         s.erase(s.begin(), std::find_if(s.begin(), s.end(), [](int ch) {
00047                 return !std::isspace(ch);
00048         }));
00049 }
00050
00051 // trim from end (in place)
00052
00053 static inline void rtrim(std::string &s) {
00054         s.erase(std::find_if(s.rbegin(), s.rend(), [](int ch) {
00055                 return !std::isspace(ch);
```

```
        }).base(), s.end());
}

// trim from both ends (in place)

static inline void trim(std::string &s) {
        ltrim(s);
        rtrim(s);
}
 */

static inline std::string strTruncIfInt(std::string strValue) {
    if (strValue.length() > 7 && strValue.substr(strValue.length() - 7, 7) == ".000000")
        return strValue.substr(0, strValue.length() - 7);
    else
        return strValue;
}
static inline std::string trim(std::string str) {
    const char* typeOfWhitespaces = " \t\n\r\f\v";
    str.erase(str.find_last_not_of(typeOfWhitespaces) + 1);
    str.erase(0, str.find_first_not_of(typeOfWhitespaces));
    return str;
}
// trim all spaces within the string (in place) -- used to transform general names into valid literals
static inline void trimwithin(std::string &str) {
    //ltrim(s);
    //rtrim(s);
    //s.erase(std::remove_if(s.begin(), s.end(), std::isspace), s.end());
    str.erase(remove(str.begin(), str.end(), ' '), str.end());
}
static inline std::string getFileName(const std::string& s) {
    char sep = '/';
    size_t i = s.rfind(sep, s.length());
    if (i != std::string::npos) {
        return (s.substr(i + 1, s.length() - i));
    }
    return s;
}
static inline std::string getPath(const std::string& s) {
    char sep = '/';
    return s.substr(0, s.find_last_of(sep));
}

/*
static std::string getRunningPath() {
        char result[ PATH_MAX ];
        ssize_t count = readlink("/proc/self/exe", result, PATH_MAX);
        std::string fullfilename = std::string(result, (count > 0) ? count : 0);
        return getPath(fullfilename);
}

static std::vector<std::string> listFiles(std::string dir, std::string fileFilter = "", mode_t
 attribFilter = S_IFREG & S_IFDIR) {
        std::vector<std::string> files;
        DIR *dp;
        struct dirent *dirp;
        struct stat statbuffer;
        if ((dp = opendir(dir.c_str())) == NULL) {
                //cout « "Error(" « errno « ") opening " « dir « endl;
                //return errno;
        }
        int status;
        while ((dirp = readdir(dp)) != NULL) {
                status = stat(dirp->d_name, &statbuffer);
                if (status & attribFilter) { //
 https://pubs.opengroup.org/onlinepubs/7908799/xsh/sysstat.h.html
                        if (fileFilter == "" || std::string(dirp->d_name).find(fileFilter) !=
 std::string::npos) {
                                files.push_back(std::string(dirp->d_name));
                        }
                }
        }
        closedir(dp);
        return files;
}
 */
class Util {
public:
    typedef unsigned long identification;
    typedef unsigned int rank;

    enum class TimeUnit : int {
        picosecond = 1,
        nanosecond = 2,
        microsecond = 3,
        milisecond = 4,
        second = 5,
```

```
        minute = 6,
        hour = 7,
        day = 8,
        week = 9
    };
    //static std::string ToStrTimeUnit(TimeUnit tu);

    enum class TraceLevel : int {
        L0_noTraces = 0,
        L1_errorFatal = 1,
        L2_results = 2,
        L3_errorRecover = 3, // or progress
        L4_warning = 4,
        L5_event = 5,
        L6_arrival = 6,
        L7_internal = 7,
        L8_detailed = 8,
        L9_mostDetailed = 9
        /*              ,

                        simulatorResult = 10,
                        toolResult = 11,
                        modelResult = 12,
                        componentResult = 13,
                        elementResult = 14,

                        modelSimulationEvent = 15,

                        componentArrival = 20,
                        simulatorInternal = 21,
                        toolInternal = 22,
                        modelSimulationInternal = 23,
                        modelInternal = 24,
                        componentInternal = 25,
                        elementInternal = 26,

                        simulatorDetailed = 30,
                        toolDetailed = 31,
                        modelSimulationDetailed = 32,
                        modelDetailed = 33,
                        componentDetailed = 34,
                        elementDetailed = 35,

                        everythingMostDetailed = 99
         */
    };
private:
    static Util::identification _S_lastId;
    static std::map<std::string, Util::identification> _S_lastIdOfType;
    static std::map<std::string, std::string> _S_TypeOf;

public: // indentation and string
    static unsigned int _S_indentation; // \todo: IT IS PRIVATE. ITS HERE JUST TO INCLUDE IT AS A WATCH
    static void SetIndent(const unsigned short indent);
    static void IncIndent();
    static void DecIndent();
    static void SepKeyVal(std::string str, std::string *key, std::string *value);
    static std::string Indent();
    static std::string SetW(std::string text, unsigned short width);
    static std::string StrTimeUnitShort(Util::TimeUnit timeUnit);
    static std::string StrTimeUnitLong(Util::TimeUnit timeUnit);
public: // identitification
    static Util::identification GenerateNewId();
    static Util::identification GenerateNewIdOfType(std::string objtype);
    static Util::identification GetLastIdOfType(std::string objtype);
    static void ResetIdOfType(std::string objtype);

public: // simulation support
    static double TimeUnitConvert(Util::TimeUnit timeUnit1, Util::TimeUnit timeUnit2);

public: // template implementations
    template<class T> static std::string TypeOf() {
        std::string name = typeid (T).name();
        std::map<std::string, std::string>::iterator it = _S_TypeOf.find(name);
        if (it != _S_TypeOf.end()) {
            return (*it).second;
        } else {
            std::string newname(name);
            while (std::isdigit(newname[0])) {
                newname.erase(0, 1);
            }
            _S_TypeOf.insert(std::pair<std::string, std::string>(name, newname));
            return newname;
        }
    }
    template<class T> static Util::identification GenerateNewIdOfType() {
```

```
        std::string objtype = Util::TypeOf<T>();
        std::map<std::string, Util::identification>::iterator it = Util::_S_lastIdOfType.find(objtype);
        if (it == Util::_S_lastIdOfType.end()) {
            // a new one. create the pair
            Util::_S_lastIdOfType.insert(std::pair<std::string, Util::identification>(objtype, 0));
            it = Util::_S_lastIdOfType.find(objtype);
        }
        Util::identification next = (*it).second;
        next++;
        (*it).second = next;
        return (*it).second;
    }

private:
    Util();
    virtual ~Util() = default;
};


// \todo Implement it using templates (check impact on calling syntax)
static inline std::string LoadField(std::map<std::string, std::string>* fields, std::string fieldName, std::string defaultValue = "") {
    return fields->find(fieldName) != fields->end() ? ((*(fields->find(fieldName))).second) : defaultValue;
}
static inline double LoadField(std::map<std::string, std::string>* fields, std::string fieldName, double defaultValue) {
    return std::stod(fields->find(fieldName) != fields->end() ? ((*(fields->find(fieldName))).second) : std::to_string(defaultValue));
}
static inline unsigned int LoadField(std::map<std::string, std::string>* fields, std::string fieldName, unsigned int defaultValue) {
    return std::stoi(fields->find(fieldName) != fields->end() ? ((*(fields->find(fieldName))).second) : std::to_string(defaultValue));
}
static inline int LoadField(std::map<std::string, std::string>* fields, std::string fieldName, int defaultValue) {
    return std::stoi(fields->find(fieldName) != fields->end() ? ((*(fields->find(fieldName))).second) : std::to_string(defaultValue));
}
static inline Util::TimeUnit LoadField(std::map<std::string, std::string>* fields, std::string fieldName, Util::TimeUnit defaultValue) {
    return static_cast<Util::TimeUnit> (std::stoi(fields->find(fieldName) != fields->end() ? ((*(fields->find(fieldName))).second) : std::to_string(static_cast<int> (defaultValue))));
}

// \todo Implement it using templates (check impact on calling syntax)
static inline void SaveField(std::map<std::string, std::string>* fields, std::string fieldName, std::string fieldValue, const std::string fieldDefaultValue = "", const bool saveIfEqualsDefault = false) {
    if (saveIfEqualsDefault || (fieldValue != fieldDefaultValue))
        fields->emplace(fieldName, "\"" + fieldValue + "\"");
}
static inline void SaveField(std::map<std::string, std::string>* fields, std::string fieldName, double fieldValue, const double fieldDefaultValue) {
    if (fieldValue != fieldDefaultValue)
        fields->emplace(fieldName, std::to_string(fieldValue));
}
static inline void SaveField(std::map<std::string, std::string>* fields, std::string fieldName, unsigned int fieldValue, const unsigned int fieldDefaultValue) {
    if (fieldValue != fieldDefaultValue)
        fields->emplace(fieldName, std::to_string(fieldValue));
}
static inline void SaveField(std::map<std::string, std::string>* fields, std::string fieldName, unsigned int fieldValue) {
    fields->emplace(fieldName, std::to_string(fieldValue));
}
static inline void SaveField(std::map<std::string, std::string>* fields, std::string fieldName, int fieldValue, const int fieldDefaultValue) {
    if (fieldValue != fieldDefaultValue)
        fields->emplace(fieldName, std::to_string(fieldValue));
}
static inline void SaveField(std::map<std::string, std::string>* fields, std::string fieldName, Util::TimeUnit fieldValue, const Util::TimeUnit fieldDefaultValue, const bool saveDefaultValue = false) {
    if (saveDefaultValue || (fieldValue != fieldDefaultValue))
        fields->emplace(fieldName, std::to_string(static_cast<int> (fieldValue)));
}

//namespace\\}
#endif /* UTIL_H */
```

8.383 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/main.cpp File Reference

`#include "Traits.h"`

Include dependency graph for main.cpp:

Functions

- int main (int argc, char ∗∗argv)

8.383.1 Function Documentation

8.383.1.1 main()

```
int main (
    int argc,
    char ** argv )
```

This is the MAIN application of GenESyS. It just calls the Application specificied on the configuration/traits file. This file should NOT be changed. Configure the application to be executed using Traits<GenesysApplication_if>

Definition at line 21 of file main.cpp.

Here is the call graph for this function:

8.384 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/main.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007
00008 /*
00009  * File:   main.cpp
00010  * Author: rafael.luiz.cancian
00011  *
00012  * Created on 21 de Junho de 2018, 12:47
00013  */
00014
00015 #include "Traits.h"
00016
00021 int main(int argc, char** argv) {
00022     // do not change it. Set you own application in Traits file =>
       Traits<GenesysApplication_if>::Application
00023     GenesysApplication_if *app = new Traits<GenesysApplication_if>::Application();
00024     int res = app->main(argc, argv);
00025     return res;
00026 }
```

8.385 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/nbproject/private/c_standard_↩ headers_indexer.c File Reference

```
#include <assert.h>
#include <ctype.h>
#include <errno.h>
#include <float.h>
#include <limits.h>
#include <locale.h>
#include <math.h>
#include <setjmp.h>
#include <signal.h>
#include <stdarg.h>
#include <stddef.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include <iso646.h>
#include <wchar.h>
#include <wctype.h>
```

Include dependency graph for c_standard_headers_indexer.c:

8.386 c_standard_headers_indexer.c

```
00001 /*
00002  * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
00003  *
00004  * Copyright (c) 2016 Oracle and/or its affiliates. All rights reserved.
00005  *
00006  * Oracle and Java are registered trademarks of Oracle and/or its affiliates.
00007  * Other names may be trademarks of their respective owners.
00008  *
00009  * The contents of this file are subject to the terms of either the GNU
00010  * General Public License Version 2 only ("GPL") or the Common
00011  * Development and Distribution License("CDDL") (collectively, the
00012  * "License"). You may not use this file except in compliance with the
00013  * License. You can obtain a copy of the License at
00014  * http://www.netbeans.org/cddl-gplv2.html
00015  * or nbbuild/licenses/CDDL-GPL-2-CP. See the License for the
00016  * specific language governing permissions and limitations under the
00017  * License.  When distributing the software, include this License Header
00018  * Notice in each file and include the License file at
00019  * nbbuild/licenses/CDDL-GPL-2-CP.  Oracle designates this
00020  * particular file as subject to the "Classpath" exception as provided
00021  * by Oracle in the GPL Version 2 section of the License file that
00022  * accompanied this code. If applicable, add the following below the
00023  * License Header, with the fields enclosed by brackets [] replaced by
00024  * your own identifying information:
00025  * "Portions Copyrighted [year] [name of copyright owner]"
00026  *
00027  * If you wish your version of this file to be governed by only the CDDL
00028  * or only the GPL Version 2, indicate your decision by adding
00029  * "[Contributor] elects to include this software in this distribution
00030  * under the [CDDL or GPL Version 2] license." If you do not indicate a
00031  * single choice of license, a recipient has the option to distribute
00032  * your version of this file under either the CDDL, the GPL Version 2 or
00033  * to extend the choice of license to its licensees as provided above.
00034  * However, if you add GPL Version 2 code and therefore, elected the GPL
00035  * Version 2 license, then the option applies only if the new code is
```

```
 * made subject to such option by the copyright holder.
 *
 * Contributor(s):
 */

// List of standard headers was taken in http://en.cppreference.com/w/c/header

#include <assert.h>   // Conditionally compiled macro that compares its argument to zero
#include <ctype.h>   // Functions to determine the type contained in character data
#include <errno.h>   // Macros reporting error conditions
#include <float.h>   // Limits of float types
#include <limits.h>   // Sizes of basic types
#include <locale.h>   // Localization utilities
#include <math.h>   // Common mathematics functions
#include <setjmp.h>   // Nonlocal jumps
#include <signal.h>   // Signal handling
#include <stdarg.h>   // Variable arguments
#include <stddef.h>   // Common macro definitions
#include <stdio.h>   // Input/output
#include <string.h>   // String handling
#include <stdlib.h>   // General utilities: memory management, program utilities, string conversions, random numbers
#include <time.h>   // Time/date utilities
#include <iso646.h>      // (since C95) Alternative operator spellings
#include <wchar.h>       // (since C95) Extended multibyte and wide character utilities
#include <wctype.h>      // (since C95) Wide character classification and mapping utilities
#ifdef _STDC_C99
#include <complex.h>     // (since C99) Complex number arithmetic
#include <fenv.h>        // (since C99) Floating-point environment
#include <inttypes.h>    // (since C99) Format conversion of integer types
#include <stdbool.h>     // (since C99) Boolean type
#include <stdint.h>      // (since C99) Fixed-width integer types
#include <tgmath.h>      // (since C99) Type-generic math (macros wrapping math.h and complex.h)
#endif
#ifdef _STDC_C11
#include <stdalign.h>    // (since C11) alignas and alignof convenience macros
#include <stdatomic.h>   // (since C11) Atomic types
#include <stdnoreturn.h> // (since C11) noreturn convenience macros
#include <threads.h>     // (since C11) Thread library
#include <uchar.h>       // (since C11) UTF-16 and UTF-32 character utilities
#endif
```

8.387 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/nbproject/private/cpp_standard_headers_indexer.cpp File Reference

```
#include <cstdlib>
#include <csignal>
#include <csetjmp>
#include <cstdarg>
#include <typeinfo>
#include <bitset>
#include <functional>
#include <utility>
#include <ctime>
#include <cstddef>
#include <new>
#include <memory>
#include <climits>
#include <cfloat>
#include <limits>
#include <exception>
#include <stdexcept>
#include <cassert>
#include <cerrno>
#include <cctype>
#include <cwctype>
#include <cstring>
#include <cwchar>
```

```
#include <string>
#include <vector>
#include <deque>
#include <list>
#include <set>
#include <map>
#include <stack>
#include <queue>
#include <algorithm>
#include <iterator>
#include <cmath>
#include <complex>
#include <valarray>
#include <numeric>
#include <iosfwd>
#include <ios>
#include <istream>
#include <ostream>
#include <iostream>
#include <fstream>
#include <sstream>
#include <strstream>
#include <iomanip>
#include <streambuf>
#include <cstdio>
#include <locale>
#include <clocale>
#include <ciso646>
```

8.388 cpp_standard_headers_indexer.cpp

```
/*
 * DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
 *
 * Copyright (c) 2016 Oracle and/or its affiliates. All rights reserved.
 *
 * Oracle and Java are registered trademarks of Oracle and/or its affiliates.
 * Other names may be trademarks of their respective owners.
 *
 * The contents of this file are subject to the terms of either the GNU
 * General Public License Version 2 only ("GPL") or the Common
 * Development and Distribution License("CDDL") (collectively, the
 * "License"). You may not use this file except in compliance with the
 * License. You can obtain a copy of the License at
 * http://www.netbeans.org/cddl-gplv2.html
 * or nbbuild/licenses/CDDL-GPL-2-CP. See the License for the
 * specific language governing permissions and limitations under the
 * License.  When distributing the software, include this License Header
 * Notice in each file and include the License file at
 * nbbuild/licenses/CDDL-GPL-2-CP.  Oracle designates this
 * particular file as subject to the "Classpath" exception as provided
 * by Oracle in the GPL Version 2 section of the License file that
 * accompanied this code. If applicable, add the following below the
 * License Header, with the fields enclosed by brackets [] replaced by
 * your own identifying information:
 * "Portions Copyrighted [year] [name of copyright owner]"
 *
 * If you wish your version of this file to be governed by only the CDDL
 * or only the GPL Version 2, indicate your decision by adding
 * "[Contributor] elects to include this software in this distribution
 * under the [CDDL or GPL Version 2] license." If you do not indicate a
 * single choice of license, a recipient has the option to distribute
 * your version of this file under either the CDDL, the GPL Version 2 or
 * to extend the choice of license to its licensees as provided above.
 * However, if you add GPL Version 2 code and therefore, elected the GPL
 * Version 2 license, then the option applies only if the new code is
 * made subject to such option by the copyright holder.
 *
 * Contributor(s):
```

```
 */

// List of standard headers was taken in http://en.cppreference.com/w/cpp/header

#include <cstdlib>      // General purpose utilities: program control, dynamic memory allocation, random numbers, sort and search
#include <csignal>      // Functions and macro constants for signal management
#include <csetjmp>      // Macro (and function) that saves (and jumps) to an execution context
#include <cstdarg>      // Handling of variable length argument lists
#include <typeinfo>      // Runtime type information utilities
#include <bitset>      // std::bitset class template
#include <functional>      // Function objects, designed for use with the standard algorithms
#include <utility>      // Various utility components
#include <ctime>      // C-style time/date utilites
#include <cstddef>      // typedefs for types such as size_t, NULL and others
#include <new>              // Low-level memory management utilities
#include <memory>      // Higher level memory management utilities
#include <climits>          // limits of integral types
#include <cfloat>      // limits of float types
#include <limits>      // standardized way to query properties of arithmetic types
#include <exception>      // Exception handling utilities
#include <stdexcept>      // Standard exception objects
#include <cassert>      // Conditionally compiled macro that compares its argument to zero
#include <cerrno>           // Macro containing the last error number
#include <cctype>           // functions to determine the type contained in character data
#include <cwctype>          // functions for determining the type of wide character data
#include <cstring>      // various narrow character string handling functions
#include <cwchar>      // various wide and multibyte string handling functions
#include <string>      // std::basic_string class template
#include <vector>      // std::vector container
#include <deque>      // std::deque container
#include <list>      // std::list container
#include <set>              // std::set and std::multiset associative containers
#include <map>              // std::map and std::multimap associative containers
#include <stack>      // std::stack container adaptor
#include <queue>      // std::queue and std::priority_queue container adaptors
#include <algorithm>      // Algorithms that operate on containers
#include <iterator>      // Container iterators
#include <cmath>            // Common mathematics functions
#include <complex>          // Complex number type
#include <valarray>         // Class for representing and manipulating arrays of values
#include <numeric>          // Numeric operations on values in containers
#include <iosfwd>           // forward declarations of all classes in the input/output library
#include <ios>              // std::ios_base class, std::basic_ios class template and several typedefs
#include <istream>          // std::basic_istream class template and several typedefs
#include <ostream>          // std::basic_ostream, std::basic_iostream class templates and several typedefs
#include <iostream>         // several standard stream objects
#include <fstream>          // std::basic_fstream, std::basic_ifstream, std::basic_ofstream class templates and several typedefs
#include <sstream>          // std::basic_stringstream, std::basic_istringstream, std::basic_ostringstream class templates and several typedefs
#include <strstream>        // std::strstream, std::istrstream, std::ostrstream(deprecated)
#include <iomanip>          // Helper functions to control the format or input and output
#include <streambuf>        // std::basic_streambuf class template
#include <cstdio>           // C-style input-output functions
#include <locale>           // Localization utilities
#include <clocale>          // C localization utilities
#include <ciso646>          // empty header. The macros that appear in iso646.h in C are keywords in C++
#if __cplusplus >= 201103L
#include <typeindex>        // (since C++11)    std::type_index
#include <type_traits>      // (since C++11)    Compile-time type information
#include <chrono>           // (since C++11)    C++ time utilites
#include <initializer_list> // (since C++11)    std::initializer_list class template
#include <tuple>            // (since C++11)    std::tuple class template
#include <scoped_allocator> // (since C++11)    Nested allocator class
#include <cstdint>          // (since C++11)    fixed-size types and limits of other types
#include <cinttypes>        // (since C++11)    formatting macros , intmax_t and uintmax_t math and conversions
#include <system_error>     // (since C++11)    defines std::error_code, a platform-dependent error code
#include <cuchar>           // (since C++11)    C-style Unicode character conversion functions
#include <array>            // (since C++11)    std::array container
#include <forward_list>     // (since C++11)    std::forward_list container
#include <unordered_set>    // (since C++11)    std::unordered_set and std::unordered_multiset unordered associative containers
#include <unordered_map>    // (since C++11)    std::unordered_map and std::unordered_multimap unordered associative containers
#include <random>           // (since C++11)    Random number generators and distributions
#include <ratio>            // (since C++11)    Compile-time rational arithmetic
#include <cfenv>            // (since C++11)    Floating-point environment access functions
#include <codecvt>          // (since C++11)    Unicode conversion facilities
#include <regex>            // (since C++11)    Classes, algorithms and iterators to support regular expression processing
#include <atomic>           // (since C++11)    Atomic operations library
#include <ccomplex>         // (since C++11)(deprecated in C++17)   simply includes the header
```

```
<complex>
#include <ctgmath>          // (since C++11)(deprecated in C++17)   simply includes the headers <ccomplex> (until C++17)<complex> (since C++17) and <cmath>: the overloads equivalent to the contents of the C header tgmath.h are already provided by those headers
#include <cstdalign>        // (since C++11)(deprecated in C++17)   defines one compatibility macro constant
#include <cstdbool>         // (since C++11)(deprecated in C++17)   defines one compatibility macro constant
#include <thread>           // (since C++11)    std::thread class and supporting functions
#include <mutex>            // (since C++11)    mutual exclusion primitives
#include <future>           // (since C++11)    primitives for asynchronous computations
#include <condition_variable> // (since C++11)  thread waiting conditions
#endif
#if __cplusplus >= 201300L
#include <shared_mutex>     // (since C++14)    shared mutual exclusion primitives
#endif
#if __cplusplus >= 201500L
#include <any>              // (since C++17)    std::any class template
#include <optional>         // (since C++17)    std::optional class template
#include <variant>          // (since C++17)    std::variant class template
#include <memory_resource>  // (since C++17)    Polymorphic allocators and memory resources
#include <string_view>      // (since C++17)    std::basic_string_view class template
#include <execution>        // (since C++17)    Predefined execution policies for parallel versions of the algorithms
#include <filesystem>       // (since C++17)    std::path class and supporting functions
#endif
```

8.389 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/Access.cpp File Reference

```
#include "Access.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Access.cpp:

8.390 Access.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Access.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:14
 */

#include "Access.h"

#include "../../kernel/simulator/Model.h"
```

```
Access::Access(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Access>(), name) {
}

std::string Access::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Access::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Access* newComponent = new Access(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Access::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Access::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Access::_initBetweenReplications() {
}

std::map<std::string, std::string>* Access::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Access::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Access::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Access>(), &Access::LoadInstance);
    // ...
    return info;
}
```

8.391 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Access.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Access.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Access

8.392 Access.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Access.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:14
 */

#ifndef ACCESS_H
#define ACCESS_H

#include "../../kernel/simulator/ModelComponent.h"

class Access : public ModelComponent {
public: // constructors
    Access(Model* model, std::string name = "");
    virtual ~Access() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* ACCESS_H */

```

8.393 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Assign.cpp File Reference

```
#include "Assign.h"
#include <string>
```

```
#include "../../kernel/simulator/Model.h"
#include "../elements/Variable.h"
#include "../../kernel/simulator/Attribute.h"
#include "../elements/Resource.h"
```

Include dependency graph for Assign.cpp:

8.394 Assign.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Assign.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 31 de Agosto de 2018, 10:10
 */

#include "Assign.h"
#include <string>
#include "../../kernel/simulator/Model.h"
#include "../elements/Variable.h"
#include "../../kernel/simulator/Attribute.h"
#include "../elements/Resource.h"

Assign::Assign(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Assign>(), name) {
}

std::string Assign::show() {
    std::string txt = ModelComponent::show() + ",assignments=[";
    for (std::list<Assignment*>::iterator it = _assignments->list()->begin(); it != _assignments->list()->end(); it++) {
        txt += (*it)->getDestination() + "=" + (*it)->getExpression() + ",";
    }
    txt = txt.substr(0, txt.length() - 1) + "]";
    return txt;
}

List<Assign::Assignment*>* Assign::getAssignments() const {
    return _assignments;
}

PluginInformation* Assign::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Assign>(), &Assign::LoadInstance);
    //info->insertDynamicLibFileDependence("attribute.so");
    info->insertDynamicLibFileDependence("variable.so");
    return info;
}

ModelComponent* Assign::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Assign* newComponent = new Assign(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Assign::_execute(Entity* entity) {
```

```
    Assignment* let;
    std::list<Assignment*>* lets = this->_assignments->list();
    for (std::list<Assignment*>::iterator it = lets->begin(); it != lets->end(); it++) {
        let = (*it);
        double value = _parentModel->parseExpression(let->getExpression());
        _parentModel->parseExpression(let->getDestination() + "=" + std::to_string(value));
        _parentModel->getTracer()->trace("Let \"" + let->getDestination() + "\" = " +
strTruncIfInt(std::to_string(value)) + "  // " + let->getExpression());
    }

    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

void Assign::_initBetweenReplications() {
}

bool Assign::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        unsigned int nv = LoadField(fields, "assignments", DEFAULT.assignmentsSize);
        for (unsigned int i = 0; i < nv; i++) {
            std::string dest = LoadField(fields, "destination" + std::to_string(i), "");
            std::string exp = LoadField(fields, "expression" + std::to_string(i), "");
            Assignment* assmt = new Assignment(dest, exp);
            this->_assignments->insert(assmt);
        }
    }
    return res;
}

std::map<std::string, std::string>* Assign::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
//Util::TypeOf<Assign>());
    Assignment* let;
    SaveField(fields, "assignments", _assignments->size(), DEFAULT.assignmentsSize);
    unsigned short i = 0;
    for (std::list<Assignment*>::iterator it = _assignments->list()->begin(); it !=
_assignments->list()->end(); it++, i++) {
        let = (*it);
        SaveField(fields, "destination" + std::to_string(i), let->getDestination(), "");
        SaveField(fields, "expression" + std::to_string(i), let->getExpression(), "");
    }
    return fields;
}

bool Assign::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: Reimplement it. Since 201910, attributes may have index, just like "atrrib1[2]" or
"att[10,1]". Because of that, the string may contain not only the name of the attribute, but also its
index and therefore, fails on the test bellow.
    for (Assignment* let : *_assignments->list()) {
        resultAll &= _parentModel->checkExpression(let->getExpression(), "assignment", errorMessage);
    }
    return resultAll;
}
```

8.395 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Assign.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Assign.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Assign
- class Assign::Assignment

8.396 Assign.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Assign.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 31 de Agosto de 2018, 10:10
00012  */
00013
00014 #ifndef ASSIGN_H
00015 #define ASSIGN_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018 #include "../../kernel/simulator/Model.h"
00019 #include "../../kernel/simulator/Plugin.h"
00020
00058 class Assign : public ModelComponent {
00059 public:
00060
00064     class Assignment {
00065     public:
00066         Assignment(std::string destination, std::string expression) {
00067             this->_destination = destination;
00068             this->_expression = expression;
00069             // an assignment is always in the form:
00070             // (destinationType) destination = expression
00071         };
00072         void setDestination(std::string _destination) {
00073             this->_destination = _destination;
```

```
        }
        std::string getDestination() const {
            return _destination;
        }
        void setExpression(std::string _expression) {
            this->_expression = _expression;
        }
        std::string getExpression() const {
            return _expression;
        }
    private:
        std::string _destination = "";
        std::string _expression = "";

    };
public:
    Assign(Model* model, std::string name = "");
    virtual ~Assign() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    List<Assignment*>* getAssignments() const;
protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:
private:

    const struct DEFAULT_VALUES {
        unsigned int assignmentsSize = 1;
    } DEFAULT;
    List<Assignment*>* _assignments = new List<Assignment*>();
};

#endif /* ASSIGN_H */

```

8.397 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/Batch.cpp File Reference

```
#include "Batch.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Batch.cpp:

8.398 Batch.cpp

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Batch.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#include "Batch.h"

#include "../../kernel/simulator/Model.h"

Batch::Batch(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Batch>(), name) {
}

std::string Batch::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Batch::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Batch* newComponent = new Batch(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Batch::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Batch::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Batch::_initBetweenReplications() {
}

std::map<std::string, std::string>* Batch::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Batch::_check(std::string* errorMessage) {
    bool resultAll = true;
    *errorMessage += "";
    // \todo: not implemented yet
    return resultAll;
}

PluginInformation* Batch::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Batch>(), &Batch::LoadInstance);
    info->insertDynamicLibFileDependence("entitygroup.so");
    return info;
}
```

8.399 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Batch.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Batch.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Batch

8.400 Batch.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Batch.h
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:14
00012  */
00013
00014 #ifndef BATCH_H
00015 #define BATCH_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018
00053 class Batch : public ModelComponent {
00054 public: // constructors
00055     Batch(Model* model, std::string name = "");
00056     virtual ~Batch() = default;
00057 public: // virtual
00058     virtual std::string show();
00059 public: // static
00060     static PluginInformation* GetPluginInformation();
00061     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00062 protected: // virtual
00063     virtual void _execute(Entity* entity);
00064     virtual void _initBetweenReplications();
```

```
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* BATCH_H */

```

8.401 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Cellular↩ Automata.cpp File Reference

```
#include "CellularAutomata.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for CellularAutomata.cpp:

8.402 CellularAutomata.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CelularAutomata.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#include "CellularAutomata.h"
#include "../../kernel/simulator/Model.h"

CelularAutomata::CelularAutomata(Model* model, std::string name) : ModelComponent(model,
    Util::TypeOf<CelularAutomata>(), name) {
}

std::string CelularAutomata::show() {
    return ModelComponent::show() + "";
}

ModelComponent* CelularAutomata::LoadInstance(Model* model, std::map<std::string, std::string>*
    fields) {
    CelularAutomata* newComponent = new CelularAutomata(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

```

```
    }
    return newComponent;
}

void CelularAutomata::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool CelularAutomata::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void CelularAutomata::_initBetweenReplications() {
}

std::map<std::string, std::string>* CelularAutomata::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool CelularAutomata::_check(std::string* errorMessage) {
    bool resultAll = true;
    *errorMessage += "";
    // \todo: not implemented yet
    return resultAll;
}

PluginInformation* CelularAutomata::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<CelularAutomata>(),
 &CelularAutomata::LoadInstance);
    info->insertDynamicLibFileDependence("entitygroup.so");
    return info;
}

```

8.403 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Cellular↩ Automata.h File Reference

```
#include <string>
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Entity.h"
```

Include dependency graph for CellularAutomata.h:

This graph shows which files directly or indirectly include this file:

Classes

- class CelularAutomata

8.404 CellularAutomata.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   CelularAutomata.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#ifndef CELULARAUTOMATA_H
#define CELULARAUTOMATA_H

#include <string>
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Entity.h"

class CelularAutomata : public ModelComponent {
public: // constructors
    CelularAutomata(Model* model, std::string name = "");
    virtual ~CelularAutomata() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};
#endif /* CELULARAUTOMATA_H */

```

8.405 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Create.cpp File Reference

```
#include "Create.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Attribute.h"
#include "Assign.h"
```

Include dependency graph for Create.cpp:

8.406 Create.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Create.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 20:12
 */

#include "Create.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Attribute.h"
#include "Assign.h"

Create::Create(Model* model, std::string name) : SourceModelComponent(model, Util::TypeOf<Create>(), name) {
    //_numberOut = new Counter(_parentModel, getName() + "." + "Count_number_in", this);
    // \todo Check if element has already been inserted and this is not needed: _parentModel->elements()->insert(_numberOut);
    _connections->setMinInputConnections(0);
    _connections->setMaxInputConnections(0);
    GetterMember getter = DefineGetterMember<SourceModelComponent>(this, &Create::getEntitiesPerCreation);
    SetterMember setter = DefineSetterMember<SourceModelComponent>(this, &Create::setEntitiesPerCreation);
    model->getControls()->insert(new SimulationControl(Util::TypeOf<Create>(), getName() + ".EntitiesPerCreation", getter, setter));
    /* \todo:
    model->getControls()->insert(new SimulationControl(Util::TypeOf<Create>(), "Time Between Creations",
            DefineGetterMember<SourceModelComponent>(this, &Create::getTimeBetweenCreationsExpression),
            DefineSetterMember<SourceModelComponent>(this, &Create::setTimeBetweenCreationsExpression))
            );
     */
}

std::string Create::show() {
    return SourceModelComponent::show();
}

void Create::_execute(Entity* entity) {
```

```
    _parentModel->getElements()->insert(entity->getClassname(), entity);
    double tnow = _parentModel->getSimulation()->getSimulatedTime();
    entity->setAttributeValue("Entity.ArrivalTime", tnow); //->find("Entity.ArrivalTime")->second->setValue(tnow);
    //entity->setAttributeValue("Entity.Picture", 1); //->find("Entity.ArrivalTime")->second->setValue(tnow);
    double timeBetweenCreations, timeScale, newArrivalTime;
    unsigned int _maxCreations = _parentModel->parseExpression(this->_maxCreationsExpression);
    if (tnow != _lastArrival) {
        _lastArrival = tnow;
        timeBetweenCreations = _parentModel->parseExpression(this->_timeBetweenCreationsExpression);
        timeScale = Util::TimeUnitConvert(this->_timeBetweenCreationsTimeUnit, _parentModel->getSimulation()->getReplicationBaseTimeUnit());
        newArrivalTime = tnow + timeBetweenCreations*timeScale;
        for (unsigned int i = 0; i<this->_entitiesPerCreation; i++) {
            if (_entitiesCreatedSoFar < _maxCreations) {
                _entitiesCreatedSoFar++;
                Entity* newEntity = _parentModel->createEntity(entity->getEntityType()->getName() + "_%", false);
                newEntity->setEntityType(entity->getEntityType());
                Event* newEvent = new Event(newArrivalTime, newEntity, this);
                _parentModel->getFutureEvents()->insert(newEvent);
                _parentModel->getTracer()->trace("Arrival of "/*entity " + std::to_string(newEntity->entityNumber())*/ + newEntity->getName() + " schedule for time " + std::to_string(newArrivalTime) + Util::StrTimeUnitShort(_parentModel->getSimulation()->getReplicationBaseTimeUnit()));
            }
        }
    }
    if (_reportStatistics)
        _numberOut->incCountValue();
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

PluginInformation* Create::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Create>(), &Create::LoadInstance);
    info->setSource(true);
    //info->insertDynamicLibFileDependence("attribute.so");
    //info->insertDynamicLibFileDependence("entitytype.so");
    //info->insertDynamicLibFileDependence("statisticscollector.so");
    return info;
}

ModelComponent* Create::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Create* newComponent = new Create(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

bool Create::_loadInstance(std::map<std::string, std::string>* fields) {
    return SourceModelComponent::_loadInstance(fields);
}

void Create::_initBetweenReplications() {
    SourceModelComponent::_initBetweenReplications();
}

std::map<std::string, std::string>* Create::_saveInstance() {
    std::map<std::string, std::string>* fields = SourceModelComponent::_saveInstance();
    return fields;
}

bool Create::_check(std::string* errorMessage) {
    bool resultAll = SourceModelComponent::_check(errorMessage);
    return resultAll;
}

void Create::_createInternalElements() {
    if (_reportStatistics && _numberOut == nullptr) {
        _numberOut = new Counter(_parentModel, getName() + "." + "CountNumberOut", this);
        _childrenElements->insert({"CountNumberOut", _numberOut});
        // \todo _childrenElements->insert("Count_number_in", _numberOut);
    } else if (!_reportStatistics && _numberOut != nullptr) {
        this->_removeChildrenElements();
        // \todo _childrenElements->remove("Count_number_in");
        //_numberOut->~Counter();
        _numberOut = nullptr;
    }
}
```

8.407 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Create.h File Reference

```
#include <string>
#include <limits>
#include "../../kernel/simulator/SourceModelComponent.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Create.h:

Classes

- class Create

8.408 Create.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Create.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 20:12
 */

#ifndef CREATE_H
#define CREATE_H

#include <string>
#include <limits>
#include "../../kernel/simulator/SourceModelComponent.h"
#include "../../kernel/simulator/EntityType.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Plugin.h"

class Create : public SourceModelComponent {
public:
    Create(Model* model, std::string name = "");
    virtual ~Create() = default;
```

```
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private:
    Counter* _numberOut = nullptr; // children element
    double _lastArrival = -1.0;
};

#endif /* CREATE_H */
```

8.409 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Decide.cpp File Reference

```
#include "Decide.h"
#include "../../kernel/simulator/Model.h"
#include "kernel/simulator/Counter.h"
```

Include dependency graph for Decide.cpp:

8.410 Decide.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Decide.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 9 de Agosto de 2018, 20:39
 */

#include "Decide.h"
#include "../../kernel/simulator/Model.h"
#include "kernel/simulator/Counter.h"

Decide::Decide(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Decide>(), name) {
}

List<std::string>* Decide::getConditions() const {
    return _conditions;
}
```

```

std::string Decide::show() {
    return ModelComponent::show() + "";
}

void Decide::_execute(Entity* entity) {
    double value;
    unsigned short i = 0;
    for (std::list<std::string>::iterator it = _conditions->list()->begin(); it != _conditions->list()->end(); it++) {
        value = _parentModel->parseExpression((*it));
        _parentModel->getTracer()->traceSimulation(_parentModel->getSimulation()->getSimulatedTime(), entity, this, std::to_string(i + 1) + "th condition evaluated to " + strTruncIfInt(std::to_string(value)) + "  // " + (*it));
        if (value) {
            if (_reportStatistics) {
                _numberOuts->getAtRank(i)->incCountValue();
            }
            _parentModel->sendEntityToComponent(entity, this->getConnections()->getConnectionAtRank(i));
            return;
        }
        i++;
    }
    _parentModel->getTracer()->traceSimulation(_parentModel->getSimulation()->getSimulatedTime(), entity, this, "No condition has been evaluated true");
    if (_reportStatistics) {
        _numberOuts->getAtRank(i)->incCountValue();
    }
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getConnectionAtRank(i));
}

void Decide::_initBetweenReplications() {
    if (_numberOuts != nullptr) {
        for (Counter* counter : *_numberOuts->list()) {
            counter->clear();
        }
    }
}

bool Decide::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        unsigned int nv = LoadField(fields, "conditions", 0);
        for (unsigned int i = 0; i < nv; i++) {
            this->_conditions->insert(LoadField(fields, "condition" + std::to_string(i), ""));
        }
    }
    return res;
}

std::map<std::string, std::string>* Decide::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance(); //Util::TypeOf<Decide>());
    SaveField(fields, "conditions", _conditions->size(), 0u);
    unsigned short i = 0;
    for (std::list<std::string>::iterator it = _conditions->list()->begin(); it != _conditions->list()->end(); it++, i++) {
        SaveField(fields, "condition" + std::to_string(i), (*it), "");
    }
    return fields;
}

bool Decide::_check(std::string* errorMessage) {
    bool allResult = true;
    std::string condition;
    for (std::list<std::string>::iterator it = _conditions->list()->begin(); it != _conditions->list()->end(); it++) {
        condition = (*it);
        allResult &= _parentModel->checkExpression(condition, "condition", errorMessage);
    }
    return allResult;
}

void Decide::_createInternalElements() {
    if (_reportStatistics && _numberOuts == nullptr) {
        _numberOuts = new List<Counter*>();
        for (unsigned int i = 0; i<this->_connections->size(); i++) {
            Counter* counter = new Counter(_parentModel, getName() + "." + "CountNumberOut" + std::to_string(i), this);
            _numberOuts->insert(counter);
            _childrenElements->insert({"CountNumberOut" + std::to_string(i), counter});
        }
    } else if (!_reportStatistics && _numberOuts != nullptr) {
        this->_removeChildrenElements();
        _numberOuts = nullptr;
    }
```

```
}

PluginInformation* Decide::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Decide>(), &Decide::LoadInstance);
    //Decide* comp = new Decide(nullptr);
    //info->setFields(comp->_saveInstance());
    //comp->~Decide();
    return info;
}

ModelComponent* Decide::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Decide* newComponent = new Decide(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}
```

8.411 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Decide.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Counter.h"
```

Include dependency graph for Decide.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Decide

8.412 Decide.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Decide.h
 * Author: rafael.luiz.cancian
 *
 * Created on 9 de Agosto de 2018, 20:39
 */

#ifndef DECIDE_H
#define DECIDE_H

#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Counter.h"

class Decide : public ModelComponent {
public:
    Decide(Model* model, std::string name = "");
    virtual ~Decide() = default;
public:
    List<std::string>* getConditions() const;

public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private:
    List<std::string>* _conditions = new List<std::string>();
    List<Counter*>* _numberOuts = nullptr;
private:

};

#endif /* DECIDE_H */

```

8.413 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Delay.cpp File Reference

```
#include "Delay.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
```

Include dependency graph for Delay.cpp:

8.414 Delay.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Delay.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 19:49
 */

#include "Delay.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"

Delay::Delay(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Delay>(), name) {

    GetterMember getter = DefineGetterMember<Delay>(this, &Delay::delay);
    SetterMember setter = DefineSetterMember<Delay>(this, &Delay::setDelay);
    model->getControls()->insert(new SimulationControl(Util::TypeOf<Delay>(), getName() + ".Delay", getter, setter));

    //GetterMember getter2 = DefineGetterMember<Delay>(this, &Delay::delayTimeUnit);
    //SetterMember setter2 = DefineSetterMember<Delay>(this, &Delay::setDelayTimeUnit);
    //model->controls()->insert(new SimulationControl(Util::TypeOf<Delay>(), getName() + ".DelayTimeUnit", getter2, setter2));


}

void Delay::setDelay(double delay) {
    _delayExpression = std::to_string(delay);
}

double Delay::delay() const {
    return _parentModel->parseExpression(_delayExpression);
}

std::string Delay::show() {
    return ModelComponent::show() +
            ",delayExpression=" + this->_delayExpression +
            ",timeUnit=" + std::to_string(static_cast<int> (this->_delayTimeUnit));
}

ModelComponent* Delay::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Delay* newComponent = new Delay(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Delay::setDelayExpression(std::string _delayExpression) {
    this->_delayExpression = _delayExpression;
}

std::string Delay::delayExpression() const {
    return _delayExpression;
}

void Delay::setDelayTimeUnit(Util::TimeUnit _delayTimeUnit) {
    this->_delayTimeUnit = _delayTimeUnit;
}

Util::TimeUnit Delay::delayTimeUnit() const {
    return _delayTimeUnit;
}

void Delay::_execute(Entity* entity) {
    double waitTime = _parentModel->parseExpression(_delayExpression);
    Util::TimeUnit stu = _parentModel->getSimulation()->getReplicationBaseTimeUnit();//getReplicationLengthTimeUnit();
    waitTime *= Util::TimeUnitConvert(_delayTimeUnit, stu);
    if (_reportStatistics) {
        _cstatWaitTime->getStatistics()->getCollector()->addValue(waitTime);
        if (entity->getEntityType()->isReportStatistics())
            entity->getEntityType()->addGetStatisticsCollector(entity->getEntityTypeName() + ".WaitTime")->getStatistics()->getCollector()->addValue(waitTime);
    }
    double totalWaitTime = entity->getAttributeValue("Entity.TotalWaitTime");
    entity->setAttributeValue("Entity.TotalWaitTime", totalWaitTime + waitTime);
```

```
    double delayEndTime = _parentModel->getSimulation()->getSimulatedTime() + waitTime;
    Event* newEvent = new Event(delayEndTime, entity, this->getConnections()->getFrontConnection());
    _parentModel->getFutureEvents()->insert(newEvent);
    _parentModel->getTracer()->trace("End of delay of "/*entity " + std::to_string(entity->entityNumber())*/ + entity->getName() + " scheduled to time " + std::to_string(delayEndTime) + Util::StrTimeUnitShort(stu) + " (wait time " + std::to_string(waitTime) + Util::StrTimeUnitShort(stu) + ") // " + _delayExpression);
}

bool Delay::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_delayExpression = LoadField(fields, "delayExpression", DEFAULT.delayExpression);
        this->_delayTimeUnit = LoadField(fields, "delayExpressionTimeUnit", DEFAULT.delayTimeUnit);
    }
    return res;
}

void Delay::_initBetweenReplications() {
}

std::map<std::string, std::string>* Delay::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance(); //Util::TypeOf<Delay>());
    SaveField(fields, "delayExpression", this->_delayExpression, DEFAULT.delayExpression);
    SaveField(fields, "delayExpressionTimeUnit", _delayTimeUnit, DEFAULT.delayTimeUnit);
    return fields;
}

bool Delay::_check(std::string* errorMessage) {
    //include attributes needed
    ElementManager* elements = _parentModel->getElements();
    std::vector<std::string> neededNames = {"Entity.TotalWaitTime"};
    std::string neededName;
    for (unsigned int i = 0; i < neededNames.size(); i++) {
        neededName = neededNames[i];
        if (elements->getElement(Util::TypeOf<Attribute>(), neededName) == nullptr) {
            Attribute* attr1 = new Attribute(_parentModel, neededName);
            elements->insert(attr1);
        }
    }
    return _parentModel->checkExpression(_delayExpression, "Delay expression", errorMessage);
}

void Delay::_createInternalElements() {
    if (_reportStatistics && _cstatWaitTime == nullptr) {
        _cstatWaitTime = new StatisticsCollector(_parentModel, getName() + "." + "WaitTime", this);
        _childrenElements->insert({"WaitTime", _cstatWaitTime});
        // include StatisticsCollector needed in EntityType
        ElementManager* elements = _parentModel->getElements();
        std::list<ModelElement*>* enttypes = elements->getElementList(Util::TypeOf<EntityType>())->list();
        for (ModelElement* element : *enttypes) {
            EntityType* enttype = static_cast<EntityType*> (element);
            if (element->isReportStatistics())
                enttype->addGetStatisticsCollector(enttype->getName() + ".WaitTime"); // force create this CStat before simulation starts
        }
    } else {
        _removeChildrenElements();
        // \todo remove StatisticsCollector needed in EntityType
    }
}

PluginInformation* Delay::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Delay>(), &Delay::LoadInstance);
    return info;
}
```

8.415 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Delay.h File Reference

```
#include <string>
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Delay.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Delay

8.416 Delay.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Delay.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Junho de 2018, 19:49
00012  */
00013
00014 #ifndef DELAY_H
00015 #define DELAY_H
00016
00017 #include <string>
00018 #include "../../kernel/simulator/ModelComponent.h"
00019 #include "../../kernel/simulator/Plugin.h"
00020
00041 class Delay : public ModelComponent {
00042 public:
00043     Delay(Model* model, std::string name = "");
00044     virtual ~Delay() = default;
00045 public:
00046     void setDelayExpression(std::string _delayExpression);
00047     std::string delayExpression() const;
00048     void setDelay(double delay);
00049     double delay() const;
00050     void setDelayTimeUnit(Util::TimeUnit _delayTimeUnit);
00051     Util::TimeUnit delayTimeUnit() const;
00052 public:
00053     virtual std::string show();
00054 public:
00055     static PluginInformation* GetPluginInformation();
00056     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00057 protected:
00058     virtual void _execute(Entity* entity);
00059     virtual void _initBetweenReplications();
00060     virtual bool _loadInstance(std::map<std::string, std::string>* fields);
00061     virtual std::map<std::string, std::string>* _saveInstance();
00062     virtual bool _check(std::string* errorMessage);
00063     virtual void _createInternalElements();
```

```
private:

    const struct DEFAULT_VALUES {
        std::string delayExpression = "1.0";
        Util::TimeUnit delayTimeUnit = Util::TimeUnit::second;
    } DEFAULT;
    std::string _delayExpression = DEFAULT.delayExpression;
    Util::TimeUnit _delayTimeUnit = DEFAULT.delayTimeUnit;
private: // inner children elements
    StatisticsCollector* _cstatWaitTime = nullptr;
};

#endif /* DELAY_H */

```

8.417 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/Dispose.cpp File Reference

```
#include "Dispose.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Dispose.cpp:

8.418 Dispose.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Dispose.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 20:13
 */

#include "Dispose.h"
#include "../../kernel/simulator/Model.h"

Dispose::Dispose(Model* model, std::string name) : SinkModelComponent(model, Util::TypeOf<Dispose>(), name) {
    //_numberOut = new Counter(_parentModel, getName() + "." + "Count_number_out", this);
    _connections->setMinOutputConnections(0);
    _connections->setMaxOutputConnections(0);
}

std::string Dispose::show() {
    return SinkModelComponent::show();
}

void Dispose::_execute(Entity* entity) {
    if (_reportStatistics) {
        _numberOut->incCountValue();
        if (entity->getEntityType()->isReportStatistics()) {
```

```
            double timeInSystem = _parentModel->getSimulation()->getSimulatedTime() - entity->getAttributeValue("Entity.ArrivalTime");
            entity->getEntityType()->addGetStatisticsCollector(entity->getEntityTypeName() + "." + "TotalTimeInSystem")->getStatistics()->getCollector()->addValue(timeInSystem);
        }
    }
    _parentModel->removeEntity(entity); //, _reportStatistics);
}

bool Dispose::_loadInstance(std::map<std::string, std::string>* fields) {
    return ModelComponent::_loadInstance(fields);
}

void Dispose::_initBetweenReplications() {
    SinkModelComponent::_initBetweenReplications();
}

std::map<std::string, std::string>* Dispose::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    return fields;

}

bool Dispose::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}

void Dispose::_createInternalElements() {
    if (_reportStatistics && _numberOut == nullptr) {
        // creates the counter (and then the CStats)
        _numberOut = new Counter(_parentModel, getName() + "." + "CountNumberIn", this);
        _childrenElements->insert({"CountNumberIn", _numberOut});
        // include StatisticsCollector needed for each EntityType
        std::list<ModelElement*>* enttypes = _parentModel->getElements()->getElementList(Util::TypeOf<EntityType>())->list();
        for (ModelElement* element : *enttypes) {
            if (element->isReportStatistics())
                static_cast<EntityType*> (element)->addGetStatisticsCollector(element->getName() + "." + "TotalTimeInSystem"); // force create this CStat before model checking
        }
    } else if (!_reportStatistics && _numberOut != nullptr) {
        //_numberOut->~Counter();
        //_numberOut = nullptr;
        // \todo: delete the CSTATS?
        _removeChildrenElements();
    }
}

PluginInformation* Dispose::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Dispose>(), &Dispose::LoadInstance);
    info->setSink(true);
    return info;
}

ModelComponent* Dispose::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Dispose* newComponent = new Dispose(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}
```

8.419 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Dispose.h File Reference

```
#include "../../kernel/simulator/SinkModelComponent.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Dispose.h:

Classes

- class Dispose

8.420 Dispose.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Dispose.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Junho de 2018, 20:13
 */

#ifndef DISPOSE_H
#define DISPOSE_H

#include "../../kernel/simulator/SinkModelComponent.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Plugin.h"

class Dispose : public SinkModelComponent {
public:
    Dispose(Model* model, std::string name = "");
    virtual ~Dispose() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private: // children elements
    Counter* _numberOut = nullptr;
};

#endif /* DISPOSE_H */

```

8.421 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/DropOff.cpp File Reference

```
#include "DropOff.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for DropOff.cpp:

8.422 DropOff.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   DropOff.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:15
 */

#include "DropOff.h"
#include "../../kernel/simulator/Model.h"

DropOff::DropOff(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<DropOff>(), name) {
}

std::string DropOff::show() {
    return ModelComponent::show() + "";
}

ModelComponent* DropOff::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    DropOff* newComponent = new DropOff(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void DropOff::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool DropOff::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}
```

```
00046
00047 void DropOff::_initBetweenReplications() {
00048 }
00049
00050 std::map<std::string, std::string>* DropOff::_saveInstance() {
00051     std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
00052     // \todo: not implemented yet
00053     return fields;
00054 }
00055
00056 bool DropOff::_check(std::string* errorMessage) {
00057     bool resultAll = true;
00058     // \todo: not implemented yet
00059     *errorMessage += "";
00060     return resultAll;
00061 }
00062
00063 PluginInformation* DropOff::GetPluginInformation() {
00064     PluginInformation* info = new PluginInformation(Util::TypeOf<DropOff>(), &DropOff::LoadInstance);
00065     // ...
00066     return info;
00067 }
00068
```

8.423 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/DropOff.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for DropOff.h:

This graph shows which files directly or indirectly include this file:

Classes

- class DropOff

8.424 DropOff.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   DropOff.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:15
 */

#ifndef DROPOFF_H
#define DROPOFF_H

#include "../../kernel/simulator/ModelComponent.h"

class DropOff : public ModelComponent {
public: // constructors
    DropOff(Model* model, std::string name = "");
    virtual ~DropOff() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* DROPOFF_H */

```

8.425 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Dummy.cpp File Reference

```
#include "Dummy.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Dummy.cpp:

8.426 Dummy.cpp

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Dummy.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Maio de 2019, 18:41
 */

#include "Dummy.h"
#include "../../kernel/simulator/Model.h"

Dummy::Dummy(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Dummy>(), name) {
}

std::string Dummy::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Dummy::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Dummy* newComponent = new Dummy(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Dummy::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Dummy::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Dummy::_initBetweenReplications() {
}

std::map<std::string, std::string>* Dummy::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Dummy::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Dummy::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Dummy>(), &Dummy::LoadInstance);
    // ...
    return info;
}
```

8.427 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Dummy.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Dummy.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Dummy

8.428 Dummy.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Dummy.h
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Maio de 2019, 18:41
 */

#ifndef DUMMY_H
#define DUMMY_H

#include "../../kernel/simulator/ModelComponent.h"

class Dummy : public ModelComponent {
public: // constructors
    Dummy(Model* model, std::string name = "");
    virtual ~Dummy() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
```

```
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};

#endif /* DUMMY_H */
```

8.429 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/Enter.cpp File Reference

```
#include "Enter.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Counter.h"
```

Include dependency graph for Enter.cpp:

8.430 Enter.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Enter.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#include "Enter.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Counter.h"

Enter::Enter(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Enter>(), name) {
}

std::string Enter::show() {
    return ModelComponent::show() + ",station=" + this->_station->getName();
}

ModelComponent* Enter::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Enter* newComponent = new Enter(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

```

```
void Enter::setStation(Station* _station) {
    this->_station = _station;
    _station->setEnterIntoStationComponent(this);
}

Station* Enter::getStation() const {
    return _station;
}

void Enter::_execute(Entity* entity) {
    if (_reportStatistics)
        _numberIn->incCountValue();
    _station->enter(entity);
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Enter::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        std::string stationName = LoadField(fields, "station", "");
        Station* station = dynamic_cast<Station*>
 (_parentModel->getElements()->getElement(Util::TypeOf<Station>(), stationName));
        this->_station = station;
    }
    return res;
}

void Enter::_initBetweenReplications() {
    _numberIn->clear();
}

std::map<std::string, std::string>* Enter::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    SaveField(fields, "station", _station->getName(), "");
    return fields;
}

bool Enter::_check(std::string* errorMessage) {
    bool resultAll = true;
    resultAll &= _parentModel->getElements()->check(Util::TypeOf<Station>(), _station, "Station",
 errorMessage);
    return resultAll;
}

PluginInformation* Enter::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Enter>(), &Enter::LoadInstance);
    info->setReceiveTransfer(true);
    info->insertDynamicLibFileDependence("station.so");
    return info;
}

void Enter::_createInternalElements() {
    if (_reportStatistics) {
        if (_numberIn == nullptr) {
            _numberIn = new Counter(_parentModel, getName() + "." + "CountNumberIn", this);
            _childrenElements->insert({"CountNumberIn", _numberIn});
        }
    } else
        if (_numberIn != nullptr) {
        _removeChildrenElements();
    }
}
```

8.431 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Enter.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Station.h"
#include "../../kernel/simulator/Counter.h"
```

Include dependency graph for Enter.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Enter

8.432 Enter.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Enter.h
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:14
00012  */
00013
00014 #ifndef ENTER_H
00015 #define ENTER_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018 #include "../elements/Station.h"
00019 #include "../../kernel/simulator/Counter.h"
00020
00099 class Enter : public ModelComponent {
00100 public:
00101     Enter(Model* model, std::string name = "");
00102     virtual ~Enter() = default;
00103 public:
00104     virtual std::string show();
00105 public:
00106     static PluginInformation* GetPluginInformation();
00107     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00108 public:
00109     void setStation(Station* _station);
00110     Station* getStation() const;
00111 protected:
```

```
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private: // association
    Station* _station;
private: // children elements
    Counter* _numberIn = nullptr;
};

#endif /* ENTER_H */

```

8.433 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Exit.cpp File Reference

```
#include "Exit.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Exit.cpp:

8.434 Exit.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Exit.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:15
 */

#include "Exit.h"

#include "../../kernel/simulator/Model.h"

Exit::Exit(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Exit>(), name) {
}

std::string Exit::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Exit::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Exit* newComponent = new Exit(model);
    try {
        newComponent->_loadInstance(fields);
```

```
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Exit::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Exit::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Exit::_initBetweenReplications() {
}

std::map<std::string, std::string>* Exit::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Exit::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Exit::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Exit>(), &Exit::LoadInstance);
    // ...
    return info;
}
```

8.435 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Exit.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Exit.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Exit

8.436 Exit.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Exit.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:15
 */

#ifndef EXIT_H
#define EXIT_H

#include "../../kernel/simulator/ModelComponent.h"

class Exit : public ModelComponent {
public: // constructors
    Exit(Model* model, std::string name = "");
    virtual ~Exit() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* EXIT_H */

```

8.437 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Hold.cpp File Reference

```
#include "Hold.h"
```

`#include "../../kernel/simulator/Model.h"`

Include dependency graph for Hold.cpp:

8.438 Hold.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Hold.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:20
00012  */
00013
00014 #include "Hold.h"
00015 #include "../../kernel/simulator/Model.h"
00016
00017 Hold::Hold(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Hold>(), name) {
00018 }
00019
00020 std::string Hold::show() {
00021     return ModelComponent::show() + "";
00022 }
00023
00024 ModelComponent* Hold::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
00025     Hold* newComponent = new Hold(model);
00026     try {
00027         newComponent->_loadInstance(fields);
00028     } catch (const std::exception& e) {
00029
00030     }
00031     return newComponent;
00032 }
00033
00034 void Hold::_execute(Entity* entity) {
00035     _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
00036     this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
00037 }
00038
00039 bool Hold::_loadInstance(std::map<std::string, std::string>* fields) {
00040     bool res = ModelComponent::_loadInstance(fields);
00041     if (res) {
00042         // \todo: not implemented yet
00043     }
00044     return res;
00045 }
00046
00047 void Hold::_initBetweenReplications() {
00048 }
00049
00050 std::map<std::string, std::string>* Hold::_saveInstance() {
00051     std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
00052     // \todo: not implemented yet
00053     return fields;
00054 }
00055
```

```
bool Hold::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Hold::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Hold>(), &Hold::LoadInstance);
    return info;
}

```

8.439 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Hold.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Hold.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Hold

8.440 Hold.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Hold.h
 * Author: rlcancian
```

```
 *
 * Created on 03 de Junho de 2019, 15:20
 */

#ifndef HOLD_H
#define HOLD_H

#include "../../kernel/simulator/ModelComponent.h"

class Hold : public ModelComponent {
public: // constructors
    Hold(Model* model, std::string name = "");
    virtual ~Hold() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* HOLD_H */

```

8.441 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Leave.cpp File Reference

```
#include "Leave.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Leave.cpp:

8.442 Leave.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Leave.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#include "Leave.h"
```

```
#include "../../kernel/simulator/Model.h"

Leave::Leave(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Leave>(), name) {
}

std::string Leave::show() {
    return ModelComponent::show() + ",station=" + this->_station->getName();
}

ModelComponent* Leave::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Leave* newComponent = new Leave(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Leave::setStation(Station* _station) {
    this->_station = _station;
}

Station* Leave::getStation() const {
    return _station;
}

void Leave::_execute(Entity* entity) {
    if (_reportStatistics)
        _numberIn->incCountValue();
    _station->leave(entity);
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Leave::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        std::string stationName = LoadField(fields, "station", "");
        Station* station = dynamic_cast<Station*>
 (_parentModel->getElements()->getElement(Util::TypeOf<Station>(), stationName));
        this->_station = station;
    }
    return res;
}

void Leave::_initBetweenReplications() {
    _numberIn->clear();
}

std::map<std::string, std::string>* Leave::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    SaveField(fields, "station", _station->getName(), "");
    return fields;
}

bool Leave::_check(std::string* errorMessage) {
    bool resultAll = true;
    resultAll &= _parentModel->getElements()->check(Util::TypeOf<Station>(), _station, "Station",
 errorMessage);
    return resultAll;
}

PluginInformation* Leave::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Leave>(), &Leave::LoadInstance);
    info->insertDynamicLibFileDependence("station.so");
    return info;
}

void Leave::_createInternalElements() {
    if (_reportStatistics) {
        if (_numberIn == nullptr) {
            _numberIn = new Counter(_parentModel, getName() + "." + "CountNumberIn", this);
            _childrenElements->insert({"CountNumberIn", _numberIn});
        } else
            if (_numberIn != nullptr) {
            _removeChildrenElements();
        }
    }
}

```

8.443 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Leave.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Station.h"
#include "../../kernel/simulator/Counter.h"
```

Include dependency graph for Leave.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Leave

8.444 Leave.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Leave.h
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:14
00012  */
00013
00014 #ifndef LEAVE_H
00015 #define LEAVE_H
```

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Station.h"
#include "../../kernel/simulator/Counter.h"

class Leave : public ModelComponent {
public:
    Leave(Model* model, std::string name = "");
    virtual ~Leave() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    void setStation(Station* _station);
    Station* getStation() const;
public:
protected:
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private: // association
    Station* _station;
private: // children elements
    Counter* _numberIn = nullptr;
};

#endif /* LEAVE_H */

```

8.445 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/LSODE.cpp File Reference

```
#include "LSODE.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for LSODE.cpp:

8.446 LSODE.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   LSODE.cpp
 * Author: rlcancian
 *
 * Created on 22 de Outubro de 2019, 22:28
 */
```

```

#include "LSODE.h"
#include "../../kernel/simulator/Model.h"

LSODE::LSODE(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<LSODE>(), name) {
}

std::string LSODE::show() {
    return ModelComponent::show() + "";
}

ModelComponent* LSODE::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    LSODE* newComponent = new LSODE(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void LSODE::setDiffEquations(Formula* formula) {
    _diffEquations = formula;
}

Formula* LSODE::getDiffEquations() const {
    return _diffEquations;
}

void LSODE::setTimeVariable(Variable* timeVariable) {
    _timeVariable = timeVariable;
}

Variable* LSODE::getTimeVariable() const {
    return _timeVariable;
}

void LSODE::setStep(double step) {
    _step = step;
}

double LSODE::getStep() const {
    return _step;
}

void LSODE::setVariables(Variable* variables) {
    _variables = variables;
}

Variable* LSODE::getVariables() const {
    return _variables;
}

bool LSODE::_doStep() {
    double initTime, time, tnow, eqResult, halfStep;
    //std::list<std::string>* eqs = _diffEquations->formulaExpressions()->list();
    unsigned int i, numEqs = _diffEquations->size();
    double k1[numEqs], k2[numEqs], k3[numEqs], k4[numEqs], valVar[numEqs];
    time = _timeVariable->value();
    initTime = time;
    tnow = _parentModel->getSimulation()->getSimulatedTime();
    bool res = time + _step <= tnow + 1e-15; // \todo: numerical error treatment by just adding 1e-15
    if (res) {
        halfStep = _step * 0.5;
        for (i = 0; i < numEqs; i++) {//(std::list<std::string>::iterator it = eqs->begin(); it !=
 eqs->end(); it++) {
            std::string expression = _diffEquations->expression(std::to_string(i));
            valVar[i] = _variables->value(std::to_string(i));
            eqResult = _parentModel->parseExpression(expression);
            k1[i] = eqResult;
        }
        time += halfStep;
        _timeVariable->setValue(time);
        for (i = 0; i < numEqs; i++) {
            _variables->setValue(std::to_string(i), valVar[i] + k1[i] * halfStep);
        }
        for (i = 0; i < numEqs; i++) {
            eqResult = _parentModel->parseExpression(_diffEquations->expression(std::to_string(i)));
            k2[i] = eqResult;
        }
        for (i = 0; i < numEqs; i++) {
            _variables->setValue(std::to_string(i), valVar[i] + k2[i] * halfStep);
        }
        for (i = 0; i < numEqs; i++) {
            eqResult = _parentModel->parseExpression(_diffEquations->expression(std::to_string(i)));
            k3[i] = eqResult;
        }
```

```
        for (i = 0; i < numEqs; i++) {
            _variables->setValue(std::to_string(i), valVar[i] + k3[i] * halfStep);
        }
        for (i = 0; i < numEqs; i++) {
            eqResult = _parentModel->parseExpression(_diffEquations->expression(std::to_string(i)));
            k4[i] = eqResult;
        }
        for (i = 0; i < numEqs; i++) {
            eqResult = _variables->value(std::to_string(i)) +(_step / 6) * (k1[i] + 2 * (k2[i] + k3[i]) + k4[i]);
            _variables->setValue(std::to_string(i), eqResult);
        }
        time = initTime + _step;
        _timeVariable->setValue(time);
    }
    return res;
}

void LSODE::_execute(Entity* entity) {
    //_parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    //for (std::list<std::string>::iterator it = _diffEquations->getFormulaExpressions()->list()->begin(); it != _diffEquations->getFormulaExpressions()->list()->end(); it++) {
    //double value = _parentModel->parseExpression((*it));
    //_parentModel->getTracer()->trace("Expression \"" + (*it) + "\" evaluates to " + std::to_string(value));
    //}
    while (_doStep()) {// execute solve ODE step by step until reach TNOW
        std::string message = "time=" + std::to_string(_timeVariable->value());
        for (unsigned int i = 0; i < _variables->dimensionSizes()->front(); i++) {
            message += " ,y[" + std::to_string(i) + "]=" + std::to_string(_variables->value(std::to_string(i)));
        }
        _parentModel->getTracer()->trace(message);
    }
    _parentModel->sendEntityToComponent(entity, getConnections()->getFrontConnection());
}

bool LSODE::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void LSODE::_initBetweenReplications() {
}

std::map<std::string, std::string>* LSODE::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool LSODE::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* LSODE::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<LSODE>(), &LSODE::LoadInstance);
    // ...
    return info;
}
```

8.447 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/LSODE.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Formula.h"
#include "../elements/Variable.h"
```

Include dependency graph for LSODE.h:

This graph shows which files directly or indirectly include this file:

Classes

- class LSODE

8.448 LSODE.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   LSODE.h
 * Author: rlcancian
 *
 * Created on 22 de Outubro de 2019, 22:28
 */

#ifndef LSODE_H
#define LSODE_H

#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Formula.h"
#include "../elements/Variable.h"

class LSODE : public ModelComponent {
public: // constructors
    LSODE(Model* model, std::string name = "");
    virtual ~LSODE() = default;
```

```
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public: // g&s
    void setDiffEquations(Formula* formula);
    Formula* getDiffEquations() const;
    void setTimeVariable(Variable* _timeVariable);
    Variable* getTimeVariable() const;
    void setStep(double _step);
    double getStep() const;
    void setVariables(Variable* _variables);
    Variable* getVariables() const;
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    //virtual void _createInternalElements();
private: // methods
    bool _doStep();
private: // attributes 1:1
    Formula* _diffEquations;
    Variable* _variables;
    Variable* _timeVariable;
    double _step;
private: // attributes 1:n
};

#endif /* LSODE_H */

```

8.449 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/MarkovChain.cpp File Reference

```
#include "MarkovChain.h"
#include "../../kernel/simulator/Model.h"
#include "../elements/Variable.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/TraitsKernel.h"
```

Include dependency graph for MarkovChain.cpp:

8.450 MarkovChain.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
```

```
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   MarkovChain.cpp
 * Author: rlcancian
 *
 * Created on 24 de Outubro de 2019, 18:26
 */

#include "MarkovChain.h"

#include "../../kernel/simulator/Model.h"
#include "../elements/Variable.h"
//#include "ProbDistribDefaultImpl1.h"
#include "../../kernel/simulator/Simulator.h"
#include "../../kernel/TraitsKernel.h"

MarkovChain::MarkovChain(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<MarkovChain>(), name) {
    _sampler = new TraitsKernel<Sampler_if>::Implementation();
}

std::string MarkovChain::show() {
    return ModelComponent::show() + "";
}

ModelComponent* MarkovChain::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    MarkovChain* newComponent = new MarkovChain(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void MarkovChain::setTransitionProbabilityMatrix(Variable* _transitionMatrix) {
    this->_transitionProbMatrix = _transitionMatrix;
}

Variable* MarkovChain::getTransitionMatrix() const {
    return _transitionProbMatrix;
}

Variable* MarkovChain::getCurrentState() const {
    return _currentState;
}

void MarkovChain::setCurrentState(Variable* _currentState) {
    this->_currentState = _currentState;
}

void MarkovChain::setInitialDistribution(Variable* _initialDistribution) {
    this->_initialDistribution = _initialDistribution;
}

Variable* MarkovChain::getInitialState() const {
    return _initialDistribution;
}

void MarkovChain::setInitilized(bool _initilized) {
    this->_initilized = _initilized;
}

bool MarkovChain::isInitilized() const {
    return _initilized;
}

void MarkovChain::_execute(Entity* entity) {
    //_parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    unsigned int size;
    double rnd, sum, value;
    if (!_initilized) {
        // define the initial state based on initial probabilities
        size = _initialDistribution->dimensionSizes()->front();
        rnd = _sampler->random(); //parentSimulator()->tools()->sampler()->random();
        double sum = 0.0;
        for (unsigned int i = 0; i < size; i++) {
            value = _initialDistribution->value(std::to_string(i));
            sum += value;
            if (sum > rnd) {
                _currentState->setValue(i); // _currentState =  i;
                break;
            }
```

```
        }
        _parentModel->getTracer()->trace("Initial current state=" + std::to_string(_currentState->value()));
        _initilized = true;
    } else {
        size = _transitionProbMatrix->dimensionSizes()->front();
        rnd = _sampler->random(); //parentSimulator()->tools()->sampler()->random();
        sum = 0.0;
        for (unsigned int i = 0; i < size; i++) {
            std::string index = std::to_string(static_cast<unsigned int> (_currentState->value())) + "," + std::to_string(i);
            value = _transitionProbMatrix->value(index);
            sum += value;
            if (sum > rnd) {
                _currentState->setValue(i);
                break;
            }
        }
        _parentModel->getTracer()->trace("Current state=" + std::to_string(_currentState->value()));
    }
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool MarkovChain::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void MarkovChain::_initBetweenReplications() {
    this->_initilized = false;
}

std::map<std::string, std::string>* MarkovChain::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool MarkovChain::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* MarkovChain::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<MarkovChain>(), &MarkovChain::LoadInstance);
    // ...
    return info;
}

```

8.451 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/MarkovChain.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Variable.h"
#include "../../kernel/statistics/Sampler_if.h"
```

Include dependency graph for MarkovChain.h:

This graph shows which files directly or indirectly include this file:

Classes

- class MarkovChain

8.452 MarkovChain.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   MarkovChain.h
00009  * Author: rlcancian
00010  *
00011  * Created on 24 de Outubro de 2019, 18:26
00012  */
00013
00014 #ifndef MARKOVCHAIN_H
00015 #define MARKOVCHAIN_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018 #include "../elements/Variable.h"
00019 #include "../../kernel/statistics/Sampler_if.h"
00020
00021 class MarkovChain : public ModelComponent {
```

```
public: // constructors
    MarkovChain(Model* model, std::string name = "");
    virtual ~MarkovChain() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public: // get and set
    void setTransitionProbabilityMatrix(Variable* _transitionMatrix);
    Variable* getTransitionMatrix() const;
    Variable* getCurrentState() const;
    void setInitialDistribution(Variable* _initialDistribution);
    Variable* getInitialState() const;
    void setInitilized(bool _initilized);
    bool isInitilized() const;
    void setCurrentState(Variable* _currentState);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
    Variable* _transitionProbMatrix;
    Variable* _initialDistribution;
    Variable* _currentState;
    bool _initilized = false;
private: // attributes 1:n
    Sampler_if* _sampler;
};

#endif /* MARKOVCHAIN_H */
```

8.453 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Match.cpp File Reference

```
#include "Match.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Match.cpp:

8.454 Match.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
```

```
 * File:   Match.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#include "Match.h"

#include "../../kernel/simulator/Model.h"

Match::Match(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Match>(), name) {
}

std::string Match::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Match::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Match* newComponent = new Match(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Match::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Match::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Match::_initBetweenReplications() {
}

std::map<std::string, std::string>* Match::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Match::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Match::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Match>(), &Match::LoadInstance);
    // ...
    return info;
}

```

8.455 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Match.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Match.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Match

8.456 Match.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Match.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#ifndef MATCH_H
#define MATCH_H

#include "../../kernel/simulator/ModelComponent.h"

class Match : public ModelComponent {
public: // constructors
    Match(Model* model, std::string name = "");
    virtual ~Match() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
```

```
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* MATCH_H */

```

8.457 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/OLD_↩ ODEelement.cpp File Reference

```
#include "OLD_ODEelement.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for OLD_ODEelement.cpp:

8.458 OLD_ODEelement.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ODE.cpp
 * Author: rlcancian
 *
 * Created on 13 de Junho de 2019, 19:12
 */

#include "OLD_ODEelement.h"
#include "../../kernel/simulator/Model.h"

OLD_ODEelement::OLD_ODEelement(Model* model, std::string name) : ModelElement(model, Util::TypeOf<OLD_ODEelement>(), name) {
    //_elems = elems;
}

std::string OLD_ODEelement::show() {
    std::string txt = ModelElement::show();
    unsigned short i = 0;
    for (std::list<ODEfunction*>::iterator it = _ODEfunctions->list()->begin(); it != _ODEfunctions->list()->end(); it++) {
        txt += ",func[" + std::to_string(i) + "]=\"" + (*it)->expression + "\",(func[" + std::to_string(i) + "][" + std::to_string((*it)->initialPoint) + "]=" + std::to_string((*it)->initialValue) + ")";
```

```
    }
    return txt;
}

double OLD_ODEelement::solve() {
    return 0.0; // dummy
}

void OLD_ODEelement::setStepH(double _h) {
    this->_stepH = _h;
}

double OLD_ODEelement::getStepH() const {
    return _stepH;
}

void OLD_ODEelement::setEndTime(double _endTime) {
    this->_endTime = _endTime;
}

double OLD_ODEelement::getEndTime() const {
    return _endTime;
}

List<ODEfunction*>* OLD_ODEelement::getODEfunctions() const {
    return _ODEfunctions;
}

PluginInformation* OLD_ODEelement::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<OLD_ODEelement>(),
 &OLD_ODEelement::LoadInstance);
    return info;
}

ModelElement* OLD_ODEelement::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    OLD_ODEelement* newElement = new OLD_ODEelement(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool OLD_ODEelement::_loadInstance(std::map<std::string, std::string>* fields) {
    return ModelElement::_loadInstance(fields);
}

std::map<std::string, std::string>* OLD_ODEelement::_saveInstance() {
    return ModelElement::_saveInstance();
}

bool OLD_ODEelement::_check(std::string* errorMessage) {
    bool result = true;
    unsigned short i = 0;
    ODEfunction* func;
    for (std::list<ODEfunction*>::iterator it = _ODEfunctions->list()->begin(); it !=
 _ODEfunctions->list()->end(); it++) {
        func = (*it);
        result &= _parentModel->checkExpression(func->expression, "expression[" + std::to_string(i++)
 + "]", errorMessage);
    }
    return result;
}
```

8.459 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/OLD_↩ ODEelement.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for OLD_ODEelement.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ODEfunction
- class OLD_ODEelement

8.460 OLD_ODEelement.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   ODE.h
 * Author: rlcancian
 *
 * Created on 13 de Junho de 2019, 19:12
 */
```

```
#ifndef OLD_ODEELEMENT_H
#define OLD_ODEELEMENT_H

#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Plugin.h"

class ODEfunction {
public:
    ODEfunction(std::string expression, double initialPoint, double initialValue) {
        this->expression = expression;
        this->initialPoint = initialPoint;
        this->initialValue = initialValue;
    }
    std::string expression;
    double initialPoint;
    double initialValue;
};

class OLD_ODEelement : public ModelElement {
public:
    OLD_ODEelement(Model* model, std::string name = "");
    virtual ~OLD_ODEelement() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    double solve();
    void setStepH(double _h);
    double getStepH() const;
    void setEndTime(double _endTime);
    double getEndTime() const;
    List<ODEfunction*>* getODEfunctions() const;
protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:

private:
    List<ODEfunction*>* _ODEfunctions = new List<ODEfunction*>();
    double _stepH = 0.1;
    double _endTime;
};

#endif /* OLD_ODEELEMENT_H */
```

8.461 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickStation.cpp File Reference

```
#include "PickStation.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for PickStation.cpp:

8.462 PickStation.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PickStation.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:08
 */

#include "PickStation.h"

#include "../../kernel/simulator/Model.h"

PickStation::PickStation(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<PickStation>(), name) {
}

std::string PickStation::show() {
    return ModelComponent::show() + "";
}

ModelComponent* PickStation::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    PickStation* newComponent = new PickStation(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void PickStation::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool PickStation::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void PickStation::_initBetweenReplications() {
}

std::map<std::string, std::string>* PickStation::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool PickStation::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* PickStation::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<PickStation>(), &PickStation::LoadInstance);
    info->insertDynamicLibFileDependence("station.so");
    return info;
}

```

8.463 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickStation.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for PickStation.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PickStation

8.464 PickStation.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PickStation.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:08
 */

#ifndef PICKSTATION_H
#define PICKSTATION_H

#include "../../kernel/simulator/ModelComponent.h"

class PickStation : public ModelComponent {
public: // constructors
    PickStation(Model* model, std::string name = "");
    virtual ~PickStation() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
```

```
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* PICKSTATION_H */

```

8.465 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickUp.cpp File Reference

```
#include "PickUp.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for PickUp.cpp:

8.466 PickUp.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PickUp.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:15
 */

#include "PickUp.h"

#include "../../kernel/simulator/Model.h"

PickUp::PickUp(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<PickUp>(), name) {
}

std::string PickUp::show() {
    return ModelComponent::show() + "";
}

ModelComponent* PickUp::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    PickUp* newComponent = new PickUp(model);
    try {
        newComponent->_loadInstance(fields);
```

```
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void PickUp::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool PickUp::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void PickUp::_initBetweenReplications() {
}

std::map<std::string, std::string>* PickUp::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool PickUp::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* PickUp::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<PickUp>(), &PickUp::LoadInstance);
    // ...
    return info;
}
```

8.467 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/PickUp.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for PickUp.h:

This graph shows which files directly or indirectly include this file:

Classes

- class PickUp

8.468 PickUp.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   PickUp.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:15
 */

#ifndef PICKUP_H
#define PICKUP_H

#include "../../kernel/simulator/ModelComponent.h"

class PickUp : public ModelComponent {
public: // constructors
    PickUp(Model* model, std::string name = "");
    virtual ~PickUp() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* PICKUP_H */

```

8.469 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Process.cpp File Reference

```
#include "Process.h"
```

Include dependency graph for Process.cpp:

8.470 Process.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Process.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Maio de 2019, 18:41
 */

#include "Process.h"
//#include "../../kernel/simulator/Model.h"

Process::Process(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Process>(), name) {
    _createInternalElements(); // its's called by the constructor because internal components can be accessed by process' public methods, so they must exist ever since
}

std::string Process::show() {
    return ModelComponent::show() + "";
}

void Process::setPriority(unsigned short _priority) {
    seize->setPriority(_priority);
}

unsigned short Process::getPriority() const {
    return seize->getPriority();
}

void Process::setAllocationType(unsigned int _allocationType) {
    seize->setAllocationType(_allocationType);
}

unsigned int Process::getAllocationType() const {
    return seize->getAllocationType();
}

List<SeizableItem*>* Process::getSeizeRequests() const {
    return seize->getSeizeRequests();
}
```

```

void Process::setQueueableItem(QueueableItem* _queueableItem) {
    seize->setQueueableItem(_queueableItem);
}

QueueableItem* Process::getQueueableItem() const {
    return seize->getQueueableItem();
}

void Process::setSaveAttribute(std::string _saveAttribute) {
    seize->setSaveAttribute(_saveAttribute);
}

std::string Process::getSaveAttribute() const {
    return seize->getSaveAttribute();
}

void Process::setDelayExpression(std::string _delayExpression) {
    delay->setDelayExpression(_delayExpression);
}

std::string Process::delayExpression() const {
    return delay->delayExpression();
}

void Process::setDelayTimeUnit(Util::TimeUnit _delayTimeUnit) {
    delay->setDelayTimeUnit(_delayTimeUnit);
}

Util::TimeUnit Process::delayTimeUnit() const {
    return delay->delayTimeUnit();
}

ModelComponent* Process::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Process* newComponent = new Process(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Process::_execute(Entity* entity) {
    this->_parentModel->sendEntityToComponent(entity, seize, 0.0);
}

bool Process::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Process::_createInternalElements() {
    if (seize == nullptr) {
        seize = new Seize(_parentModel, getName() + ".Seize");
        delay = new Delay(_parentModel, getName() + ".Delay");
        release = new Release(_parentModel, getName() + ".Release");
        seize->getConnections()->insert(delay);
        delay->getConnections()->insert(release);
        _childrenElements->insert({"Seize", seize});
        _childrenElements->insert({"Delay", delay});
        _childrenElements->insert({"Release", release});
    }
    release->getConnections()->list()->clear();
    if (getConnections()->size() > 0 && getConnections()->front() != seize) {
        // chance connections so Process is connected to Seize, and Release to the one Process was connected to
        _nextComponent = getConnections()->getFrontConnection()->first;
        _nextInput = getConnections()->getFrontConnection()->second;
        release->getConnections()->insert(_nextComponent, _nextInput);
        getConnections()->getFrontConnection()->first = seize;
        getConnections()->getFrontConnection()->second = 0;
    }
}

std::map<std::string, std::string>* Process::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Process::_check(std::string* errorMessage) {
    bool resultAll = true;
    // garantee that release releases exactlly whay seize seizes
```

```
    release->getReleaseRequests()->clear();
    for (SeizableItem* item : *seize->getSeizeRequests()->list()) {
        release->getReleaseRequests()->insert(item);
    }
    resultAll &= ModelComponent::Check(seize);
    resultAll &= ModelComponent::Check(delay);
    resultAll &= ModelComponent::Check(release);
    //*errorMessage += "";
    return resultAll;
}

PluginInformation* Process::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Process>(), &Process::LoadInstance);
    info->insertDynamicLibFileDependence("seize.so");
    info->insertDynamicLibFileDependence("delay.so");
    info->insertDynamicLibFileDependence("release.so");
    return info;
}
```

8.471 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Process.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "Seize.h"
#include "Delay.h"
#include "Release.h"
```

Include dependency graph for Process.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Process

8.472 Process.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Process.h
 * Author: rafael.luiz.cancian
 *
 * Created on 22 de Maio de 2019, 18:41
 */

#ifndef PROCESS_H
#define PROCESS_H

#include "../../kernel/simulator/ModelComponent.h"
#include "Seize.h"
#include "Delay.h"
#include "Release.h"

class Process : public ModelComponent {
public: // constructors
    Process(Model* model, std::string name = "");
    virtual ~Process() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);

public:

    void setPriority(unsigned short _priority);
    unsigned short getPriority() const;
    void setAllocationType(unsigned int _allocationType);
    unsigned int getAllocationType() const;
    List<SeizableItem*>* getSeizeRequests() const;
    void setQueueableItem(QueueableItem* _queueableItem);
    QueueableItem* getQueueableItem() const;
    void setSaveAttribute(std::string _saveAttribute);
    std::string getSaveAttribute() const;
    void setDelayExpression(std::string _delayExpression);
    std::string delayExpression() const;
    void setDelayTimeUnit(Util::TimeUnit _delayTimeUnit);
    Util::TimeUnit delayTimeUnit() const;



protected: // virtual
    virtual void _execute(Entity* entity);
    // virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private: // methods
private: // attributes 1:1
    Seize* seize;
    Delay* delay;
    Release* release;
    ModelComponent* _nextComponent;
    unsigned int _nextInput;
private: // attributes 1:n
};

#endif /* PROCESS_H */
```

8.473 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Queueable↩ Item.cpp File Reference

```
#include "QueueableItem.h"
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/Model.h"
```

#include <cassert>

Include dependency graph for QueueableItem.cpp:

8.474 QueueableItem.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   QueueableItem.cpp
 * Author: rlcancian
 *
 * Created on 23 de abril de 2021, 15:09
 */

#include "QueueableItem.h"
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/Model.h"
#include <cassert>

QueueableItem::QueueableItem(ModelElement* queueOrSet, QueueableItem::QueueableType queueableType, std::string index) {
    _queueableType = queueableType;
    _queueOrSet = queueOrSet;
    _index = index;
}

/* Queue* QueueableItem::getRequestedQueue() const {
    if (_queueableType == QueueableItem::QueueableType::QUEUE) {
        return static_cast<Queue*> (_queueOrSet);
    } else {
        // assume it is a set ==> _queueableType == QueueableItem::QueueableType::SET
        unsigned int index = this-> ??? ->parse(...)
    }
} */
bool QueueableItem::loadInstance(std::map<std::string, std::string>* fields) {
    bool res = true;
    try {
        _queueableType = static_cast<QueueableItem::QueueableType> (LoadField(fields, "queueableType", static_cast<int> (DEFAULT.queueableType)));
        _queueableName = LoadField(fields, "queueable", "");
        _index = LoadField(fields, "index", DEFAULT.index);
        if (_elementManager != nullptr) {
            if (_queueableType == QueueableItem::QueueableType::QUEUE) {
                _queueOrSet = _elementManager->getElement(Util::TypeOf<Queue>(), _queueableName);
            } else if (_queueableType == QueueableItem::QueueableType::SET) {
                _queueOrSet = _elementManager->getElement(Util::TypeOf<Queue>(), _queueableName);
                //            } else if (_queueableType == QueueableItem::QueueableType::HOLD) {
                //                _queueOrSet = _elementManager->getElement(Util::TypeOf<Hold>(), _queueableName);
            }
            assert(_queueOrSet != nullptr);
        }
    } catch (...) {
        res = false;
    }
    return res;
```

```
}

std::map<std::string, std::string>* QueueableItem::saveInstance() {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    SaveField(fields, "queueableType", static_cast<int> (_queueableType), static_cast<int> (DEFAULT.queueableType));
    SaveField(fields, "queueId", _queueOrSet->getId());
    SaveField(fields, "queueable", _queueOrSet->getName(), "");
    SaveField(fields, "index", _index, DEFAULT.index);
    return fields;
}

/*
bool QueueableItem::_loadInstance(std::map<std::string, std::string>* fields, unsigned int parentIndex) {
    bool res = true;
    std::string num = std::to_string(parentIndex);
    try {
        _queueableType = static_cast<QueueableItem::QueueableType> (LoadField(fields, "queueableType" + num, static_cast<int> (DEFAULT.queueableType)));
        _queueableName = LoadField(fields, "queueable" + num, "");
        _index = LoadField(fields, "index" + num, DEFAULT.index);
    } catch (...) {
        res = false;
    }
    return res;
}

std::map<std::string, std::string>* QueueableItem::_saveInstance(unsigned int parentIndex) {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    std::string num = std::to_string(parentIndex);
    SaveField(fields, "queueableType" + num, static_cast<int> (_queueableType), static_cast<int> (DEFAULT.queueableType));
    //SaveField(fields, "queueId" + num, _queueOrSet->getId());
    SaveField(fields, "queueable" + num, _queueOrSet->getName(), "");
    SaveField(fields, "index" + num, _index, DEFAULT.index);
    return fields;
}
 */


std::string QueueableItem::show() {
    return "queueType=" + std::to_string(static_cast<int> (_queueableType)) + ",queue=\"" + _queueOrSet->getName() + "\",index=\"" + _index + "\"";
}

void QueueableItem::setIndex(std::string index) {
    this->_index = index;
}

std::string QueueableItem::getIndex() const {
    return _index;
}

std::string QueueableItem::getQueueableName() const {
    return _queueableName;
}

void QueueableItem::setQueue(Queue* queue) {
    this->_queueOrSet = queue;
    _queueableName = queue->getName();
}

Queue* QueueableItem::getQueue() const {
    return static_cast<Queue*> (_queueOrSet);
}

void QueueableItem::setSet(Set* set) {
    this->_queueOrSet = set;
    _queueableName = set->getName();
}

Set* QueueableItem::getSet() const {
    return static_cast<Set*> (_queueOrSet);
}

void QueueableItem::setQueueableType(QueueableItem::QueueableType queueableType) {
    this->_queueableType = queueableType;
}

QueueableItem::QueueableType QueueableItem::getQueueableType() const {
    return _queueableType;
}

ModelElement* QueueableItem::getQueueable() const {
    return _queueOrSet;
}
```

```
00135
00136 void QueueableItem::setElementManager(ElementManager* _elementManager) {
00137     this->_elementManager = _elementManager;
00138 }
00139
00140 //void QueueableItem::setComponentManager(ComponentManager* componentManager) {
00141 //    this->_componentManager = componentManager;
00142 //}
```

8.475 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/QueueableItem.h File Reference

```
#include "../elements/Queue.h"
#include "../elements/Set.h"
#include "../../kernel/simulator/ElementManager.h"
```

Include dependency graph for QueueableItem.h:

This graph shows which files directly or indirectly include this file:

Classes

- class QueueableItem

8.476 QueueableItem.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   QueueableItem.h
```

```
 * Author: rlcancian
 *
 * Created on 23 de abril de 2021, 15:09
 */

#ifndef QUEUEABLEITEM_H
#define QUEUEABLEITEM_H

#include "../elements/Queue.h"
#include "../elements/Set.h"
#include "../../kernel/simulator/ElementManager.h"

// \todo should inhere from a common base to SeizeableItem

class QueueableItem {
public:

    enum class QueueableType : int {
        QUEUE = 1, SET = 2//, HOLD = 3
    };

public:
    QueueableItem(ModelElement* queueOrSet, QueueableItem::QueueableType queueableType = QueueableItem::QueueableType::QUEUE, std::string index = "0");
public:
    //virtual bool _loadInstance(std::map<std::string, std::string>* fields, unsigned int parentIndex);
    //virtual std::map<std::string, std::string>* _saveInstance(unsigned int parentIndex);
    bool loadInstance(std::map<std::string, std::string>* fields);
    std::map<std::string, std::string>* saveInstance();
public:
    std::string show();
    void setIndex(std::string index);
    std::string getIndex() const;
    std::string getQueueableName() const;
    void setQueue(Queue* resource);
    Queue* getQueue() const;
    void setSet(Set* set);
    Set* getSet() const;
    //
    void setQueueableType(QueueableType queueableType);
    QueueableType getQueueableType() const;
    ModelElement* getQueueable() const;
    void setElementManager(ElementManager* _elementManager);
    //void setComponentManager(ComponentManager* _componentManager);

private:

    const struct DEFAULT_VALUES {
        const std::string quantityExpression = "1";
        const QueueableItem::QueueableType queueableType = QueueableItem::QueueableType::QUEUE;
        const std::string saveAttribute = "";
        const std::string index = "0";
    } DEFAULT;

    QueueableType _queueableType;
    ModelElement* _queueOrSet;
    std::string _queueableName;
    std::string _index;
private:
    //ComponentManager* _componentManager;
    ElementManager* _elementManager;
};


#endif /* QUEUEABLEITEM_H */

```

8.477 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Record.cpp File Reference

```
#include "Record.h"
#include "../../kernel/simulator/Model.h"
#include <fstream>
#include <cstdio>
```

#include <iostream>

Include dependency graph for Record.cpp:

8.478 Record.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Record.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 9 de Agosto de 2018, 13:52
 */

#include "Record.h"
#include "../../kernel/simulator/Model.h"
#include <fstream>
#include <cstdio>
#include <iostream>

Record::Record(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Record>(), name) {
}

Record::~Record() {
    _parentModel->getElements()->remove(Util::TypeOf<StatisticsCollector>(), _cstatExpression);
}

std::string Record::show() {
    return ModelComponent::show() +
            ",expressionName=\"" + this->_expressionName + "\"" +
            ",expression=\"" + _expression + "\"" +
            "filename=\"" + _filename + "\"";
}

void Record::setExpressionName(std::string expressionName) {
    this->_expressionName = expressionName;
    if (_cstatExpression != nullptr)
        this->_cstatExpression->setName(getName() + "." + expressionName);
}

std::string Record::getExpressionName() const {
    return _expressionName;
}

StatisticsCollector* Record::getCstatExpression() const {
    return _cstatExpression;
}

void Record::setFilename(std::string filename) {
    this->_filename = filename;
}

std::string Record::getFilename() const {
    return _filename;
}
```

```cpp

void Record::setExpression(const std::string expression) {
    this->_expression = expression;
}

std::string Record::getExpression() const {
    return _expression;
}

void Record::_execute(Entity* entity) {
    double value = _parentModel->parseExpression(_expression);
    _cstatExpression->getStatistics()->getCollector()->addValue(value);
    if (_filename != "") {
        // \todo: open and close for every data is not a good idea. Should open when replication starts and close when it finishes.
        std::ofstream file;
        file.open(_filename, std::ofstream::out | std::ofstream::app);
        file « value « std::endl;
        file.close();
    }
    _parentModel->getTracer()->traceSimulation(_parentModel->getSimulation()->getSimulatedTime(), entity, this, "Recording value " + std::to_string(value));
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());

}

std::map<std::string, std::string>* Record::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance(); //Util::TypeOf<Record>());
    SaveField(fields, "expression0", this->_expression, "");
    SaveField(fields, "expressionName0", this->_expressionName, "");
    SaveField(fields, "fileName0", this->_filename, "");
    return fields;
}

bool Record::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_expression = LoadField(fields, "expression0", "");
        this->_expressionName = LoadField(fields, "expressionName0", "");
        this->_filename = LoadField(fields, "fileName0", "");
    }
    return res;
}

void Record::_initBetweenReplications() {
}

bool Record::_check(std::string* errorMessage) {
    // when cheking the model (before simulating it), remove the file if exists
    std::remove(_filename.c_str());
    return _parentModel->checkExpression(_expression, "expression", errorMessage);
}

void Record::_createInternalElements() {
    if (_reportStatistics && _cstatExpression == nullptr) {
        _cstatExpression = new StatisticsCollector(_parentModel, getName() + "." + _expressionName, this);
        //_parentModel->getElements()->insert(_cstatExpression);
        _childrenElements->insert({_expressionName, _cstatExpression});
    } else if (!_reportStatistics && _cstatExpression != nullptr) {
        this->_removeChildrenElements();
        _cstatExpression = nullptr;
    }
}

PluginInformation* Record::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Record>(), &Record::LoadInstance);
    return info;
}

ModelComponent* Record::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Record* newComponent = new Record(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;

}

```

8.479 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Record.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include <string>
```

Include dependency graph for Record.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Record

8.480 Record.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Record.h
 * Author: rafael.luiz.cancian
 *
 * Created on 9 de Agosto de 2018, 13:52
 */

#ifndef RECORD_H
#define RECORD_H

#include "../../kernel/simulator/ModelComponent.h"
#include <string>

class Record : public ModelComponent {
public:
    Record(Model* model, std::string name = "");
    virtual ~Record();
public:
    void setFilename(std::string filename);
    std::string getFilename() const;
    void setExpression(const std::string expression);
    std::string getExpression() const;
    void setExpressionName(std::string expressionName);
```

```
    std::string getExpressionName() const;
    StatisticsCollector* getCstatExpression() const;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private:
    std::string _expression = "";
    std::string _expressionName = "";
    std::string _filename = "";
private:
    // not a child element
    StatisticsCollector* _cstatExpression; /* \todo: Cretae an internal class to agregate
 ExpressionStatisticsColelctor, and change Record to got a list of it, so Record cn record a set of
 expressions into a set of files */
};

#endif /* RECORD_H */

```

8.481 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Release.cpp File Reference

```
#include "Release.h"
#include "../../kernel/simulator/Model.h"
#include "../elements/Resource.h"
#include "../../kernel/simulator/Attribute.h"
#include <assert.h>
#include <cmath>
```

Include dependency graph for Release.cpp:

8.482 Release.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Release.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Agosto de 2018, 16:17
 */

#include "Release.h"
```

```
#include "../../kernel/simulator/Model.h"
#include "../elements/Resource.h"
#include "../../kernel/simulator/Attribute.h"
#include <assert.h>
#include <cmath>

Release::Release(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Release>(), name) {
}

std::string Release::show() {
    std::string txt = ModelComponent::show() +
            "priority=" + std::to_string(_priority) +
            "releaseRequests={";
    unsigned short i = 0;
    for (SeizableItem* item : *_releaseRequests->list()) {
        txt += "request" + std::to_string(i++) + "=[" + item->show() + "],";
    }
    txt = txt.substr(0, txt.length() - 1) + "}";
    return txt;
}

void Release::setPriority(unsigned short _priority) {
    this->_priority = _priority;
}

unsigned short Release::priority() const {
    return _priority;
}

List<SeizableItem*>* Release::getReleaseRequests() const {
    return _releaseRequests;
}

//void Release::setResource(Resource* _resource) {
//  this->_resource = _resource;
//}

//Resource* Release::resource() const {
//  return _resource;
//}

void Release::_execute(Entity* entity) {
    for (SeizableItem* seizable : *_releaseRequests->list()) {
        Resource* resource;
        if (seizable->getSeizableType() == SeizableItem::SeizableType::RESOURCE) {
            resource = seizable->getResource();
        } else { // assume SET
            SeizableItem::SelectionRule rule = seizable->getSelectionRule();
            Set* set = seizable->getSet();
            unsigned int index = 0;
            switch (rule) { // \todo: Rules for release are DIFFERENT (least and first member seized)
                case SeizableItem::SelectionRule::CYCLICAL:
                    index = (seizable->getLastMemberSeized() + 1) % _releaseRequests->list()->size();
                    break;
                case SeizableItem::SelectionRule::LARGESTREMAININGCAPACITY:
                    // \todo
                    break;
                case SeizableItem::SelectionRule::RANDOM:
                    index = trunc(rand() * _releaseRequests->list()->size());
                    break;
                case SeizableItem::SelectionRule::SMALLESTNUMBERBUSY:
                    // \todo
                    break;
                case SeizableItem::SelectionRule::SPECIFICMEMBER:
                    index = _parentModel->parseExpression(seizable->getIndex());
                    break;
            }
            _parentModel->getTracer()->trace("Member of set " + set->getName() + " chosen index " + std::to_string(index), Util::TraceLevel::L8_detailed);
            resource = static_cast<Resource*> (set->getElementSet()->getAtRank(index));
            assert(resource != nullptr);
        }
        unsigned int quantity = _parentModel->parseExpression(seizable->getQuantityExpression());
        assert(resource->getNumberBusy() >= quantity); // 202104 ops. maybe not anymore
        _parentModel->getTracer()->traceSimulation(_parentModel->getSimulation()->getSimulatedTime(), entity, this, entity->getName() + " releases " + std::to_string(quantity) + " units of resource \"" + resource->getName() + "\" seized on time " + std::to_string(resource->getLastTimeSeized()));
        resource->release(quantity, _parentModel->getSimulation()->getSimulatedTime()); //{releases and sets the 'LastTimeSeized'property}
    }
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

void Release::_initBetweenReplications() {
    //for (SeizableItem* seizable : *_releaseRequests->list()) {
    //  if (seizable->getSeizableType() == SeizableItem::SeizableType::RESOURCE)
```

```
    //      seizable->getResource()->initBetweenReplications();
    //  else if (seizable->getSeizableType() == SeizableItem::SeizableType::SET)
    //      seizable->getSet()->initBetweenReplications();
    //}
}

bool Release::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_priority = std::stoi(LoadField(fields, "priority", "0"));
        unsigned short numRequests = LoadField(fields, "resquestSize", DEFAULT.releaseRequestSize);
        for (unsigned short i = 0; i < numRequests; i++) {
            SeizableItem* Item = new SeizableItem(nullptr);
            Item->setElementManager(_parentModel->getElements());
            Item->loadInstance(fields, i);
            //Resource* resource = static_cast<Resource*>(_parentModel->getElements()->getElement(Util::TypeOf<Resource>(), Item->getResourceName()));
            //Item->setResource(resource);
            //this->_releaseRequests->insert(Item);
        }
    }
    return res;
}

std::map<std::string, std::string>* Release::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance(); //Util::TypeOf<Release>());
    if (_priority != 0) SaveField(fields, "priority", std::to_string(this->_priority));
    SaveField(fields, "resquestSize", _releaseRequests->size(), DEFAULT.releaseRequestSize);
    unsigned short i = 0;
    for (SeizableItem* request : *_releaseRequests->list()) {
        std::map<std::string, std::string>* seizablefields = request->saveInstance(i);
        fields->insert(seizablefields->begin(), seizablefields->end());
        i++;
    }
    return fields;
}

bool Release::_check(std::string* errorMessage) {
    bool resultAll = true;

    for (SeizableItem* seizable : * _releaseRequests->list()) {
        resultAll &= _parentModel->checkExpression(seizable->getQuantityExpression(), "quantity", errorMessage);
        if (seizable->getSeizableType() == SeizableItem::SeizableType::RESOURCE) {
            resultAll &= _parentModel->getElements()->check(Util::TypeOf<Resource>(), seizable->getResource(), "Resource", errorMessage);
        } else if (seizable->getSeizableType() == SeizableItem::SeizableType::SET) {
            resultAll &= _parentModel->getElements()->check(Util::TypeOf<Set>(), seizable->getSet(), "Set", errorMessage);
        }
        // \todo: Should be checking saveAttribute, index, etc
    }
    return resultAll;
}

//void Release::setResourceName(std::string resourceName) throw () {
//  ModelElement* resource = _parentModel->elements()->element(Util::TypeOf<Resource>(), resourceName);
//  if (resource != nullptr) {
//      this->_resource = dynamic_cast<Resource*> (resource);
//  } else {
//      throw std::invalid_argument("Resource does not exist");
//  }
//}

//std::string Release::resourceName() const {
//  return _resource->name();
//}

PluginInformation* Release::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Release>(), &Release::LoadInstance);
    info->insertDynamicLibFileDependence("resource.so");
    return info;
}

ModelComponent* Release::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Release* newComponent = new Release(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;

}
```

8.483 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Release.h File Reference

```
#include <string>
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Plugin.h"
#include "SeizableItem.h"
#include "../elements/Resource.h"
```

Include dependency graph for Release.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Release

8.484 Release.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Release.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Agosto de 2018, 16:17
00012  */
00013
00014 #ifndef RELEASE_H
00015 #define RELEASE_H
00016
00017 #include <string>
00018
00019 #include "../../kernel/simulator/ModelComponent.h"
00020 #include "../../kernel/simulator/Plugin.h"
00021 #include "SeizableItem.h"
00022 #include "../elements/Resource.h"
```

```

class Release : public ModelComponent {
public:
    Release(Model* model, std::string name = "");
    virtual ~Release() = default;
public:
    virtual std::string show();
public: //static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public: // get & set
    void setPriority(unsigned short _priority);
    unsigned short priority() const;
public: // gets
    List<SeizableItem*>* getReleaseRequests() const;

protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:

    const struct DEFAULT_VALUES {
        unsigned short priority = 0;
        unsigned int releaseRequestSize = 1;
    } DEFAULT;
    unsigned short _priority = DEFAULT.priority;
    List<SeizableItem*>* _releaseRequests = new List<SeizableItem*>();
};

#endif /* RELEASE_H */

```

8.485 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/components/Remove.cpp File Reference

```
#include "Remove.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Remove.cpp:

8.486 Remove.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Remove.cpp
```

```
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:20
 */

#include "Remove.h"
#include "../../kernel/simulator/Model.h"

Remove::Remove(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Remove>(), name) {
}

std::string Remove::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Remove::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Remove* newComponent = new Remove(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Remove::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Remove::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Remove::_initBetweenReplications() {
}

std::map<std::string, std::string>* Remove::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Remove::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Remove::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Remove>(), &Remove::LoadInstance);
    // ...
    return info;
}
```

8.487 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Remove.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Remove.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Remove

8.488 Remove.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Remove.h
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:20
00012  */
00013
00014 #ifndef REMOVE_H
00015 #define REMOVE_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018
00037 class Remove : public ModelComponent {
00038 public: // constructors
00039     Remove(Model* model, std::string name = "");
00040     virtual ~Remove() = default;
00041 public: // virtual
00042     virtual std::string show();
00043 public: // static
00044     static PluginInformation* GetPluginInformation();
00045     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00046 protected: // virtual
```

```
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* REMOVE_H */

```

8.489 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Route.cpp File Reference

```
#include <cassert>
#include "Route.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../kernel/simulator/Simulator.h"
#include "../elements/Sequence.h"
```

Include dependency graph for Route.cpp:

8.490 Route.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Route.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:15
 */

#include <cassert>
#include "Route.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../kernel/simulator/Simulator.h"
#include "../elements/Sequence.h"

Route::Route(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Route>(), name) {
}

std::string Route::show() {
    std::string msg = ModelComponent::show() +
            ",destinationType=" + std::to_string(static_cast<int> (this->_routeDestinationType)) +
```

```
            ",timeExpression=" + this->_routeTimeExpression + " " + Util::StrTimeUnitShort(this->_routeTimeTimeUnit);
    if (_station != nullptr)
        msg += ",station=" + this->_station->getName();
    return msg;
}

ModelComponent* Route::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Route* newComponent = new Route(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Route::setStation(Station* _station) {
    this->_station = _station;
}

Station* Route::getStation() const {
    return _station;
}

void Route::setRouteTimeExpression(std::string _routeTimeExpression) {
    this->_routeTimeExpression = _routeTimeExpression;
}

std::string Route::getRouteTimeExpression() const {
    return _routeTimeExpression;
}

void Route::setRouteTimeTimeUnit(Util::TimeUnit _routeTimeTimeUnit) {
    this->_routeTimeTimeUnit = _routeTimeTimeUnit;
}

Util::TimeUnit Route::getRouteTimeTimeUnit() const {
    return _routeTimeTimeUnit;
}

void Route::setRouteDestinationType(DestinationType _routeDestinationType) {
    this->_routeDestinationType = _routeDestinationType;
}

Route::DestinationType Route::getRouteDestinationType() const {
    return _routeDestinationType;
}

void Route::_execute(Entity* entity) {
    Station* destinyStation = _station;
    if (_routeDestinationType == Route::DestinationType::BySequence) {
        unsigned int sequenceId = static_cast<unsigned int>(entity->getAttributeValue("Entity.Sequence"));
        unsigned int step = static_cast<unsigned int>(entity->getAttributeValue("Entity.SequenceStep"));
        Sequence* sequence = static_cast<Sequence*>(_parentModel->getElements()->getElement(Util::TypeOf<Sequence>(), sequenceId));
        SequenceStep* seqStep = sequence->getSteps()->getAtRank(step);
        if (seqStep == nullptr) {
            step = 0;
            seqStep = sequence->getSteps()->getAtRank(step);
            assert(seqStep != nullptr);
        }
        destinyStation = seqStep->getStation();
        for (SequenceStep::Assignment* assignment : *seqStep->getAssignments()) {
            _parentModel->parseExpression(assignment->getDestination() + "=" + assignment->getExpression());
        }
        entity->setAttributeValue("Entity.SequenceStep", step + 1.0);
    }
    if (_reportStatistics)
        _numberIn->incCountValue();
    // adds the route time to the TransferTime statistics / attribute related to the Entitys
    double routeTime = _parentModel->parseExpression(_routeTimeExpression) * Util::TimeUnitConvert(_routeTimeTimeUnit, _parentModel->getSimulation()->getReplicationBaseTimeUnit());
    if (entity->getEntityType()->isReportStatistics())
        entity->getEntityType()->addGetStatisticsCollector(entity->getEntityTypeName() + ".TransferTime")->getStatistics()->getCollector()->addValue(routeTime);
    entity->setAttributeValue("Entity.TotalTransferTime", entity->getAttributeValue("Entity.TotalTransferTime") + routeTime);
    if (routeTime > 0.0) {
        // calculates when this Entity will reach the end of this route and schedule this Event
        double routeEndTime = _parentModel->getSimulation()->getSimulatedTime() + routeTime;
        Event* newEvent = new Event(routeEndTime, entity, destinyStation->getEnterIntoStationComponent());
```

```
        _parentModel->getFutureEvents()->insert(newEvent);
        _parentModel->getTracer()->trace("End of route of "/*entity " + std::to_string(entity->entityNumber())*/ + entity->getName() + " to the component \"" + destinyStation->getEnterIntoStationComponent()->getName() + "\" was scheduled to time " + std::to_string(routeEndTime));
    } else {
        // send without delay
        _parentModel->sendEntityToComponent(entity, destinyStation->getEnterIntoStationComponent(), 0.0);
    }
}

bool Route::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_routeTimeExpression = LoadField(fields, "routeTimeExpression", DEFAULT.routeTimeExpression);
        this->_routeTimeTimeUnit = LoadField(fields, "routeTimeTimeUnit", DEFAULT.routeTimeTimeUnit);
        this->_routeDestinationType = static_cast<Route::DestinationType> (LoadField(fields, "destinationType", static_cast<int> (DEFAULT.routeDestinationType)));
        if (_routeDestinationType == DestinationType::Station) {
            std::string stationName = LoadField(fields, "station", "");
            Station* station = dynamic_cast<Station*> (_parentModel->getElements()->getElement(Util::TypeOf<Station>(), stationName));
            this->_station = station;
        }
    }
    return res;
}

void Route::_initBetweenReplications() {
    _numberIn->clear();
}

std::map<std::string, std::string>* Route::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    if (_routeDestinationType == DestinationType::Station) {
        SaveField(fields, "station", (this->_station->getName()));
    }
    SaveField(fields, "routeTimeExpression", _routeTimeExpression, DEFAULT.routeTimeExpression);
    SaveField(fields, "routeTimeTimeUnit", _routeTimeTimeUnit, DEFAULT.routeTimeTimeUnit);
    SaveField(fields, "destinationType", static_cast<int> (_routeDestinationType), static_cast<int> (DEFAULT.routeDestinationType));
    return fields;
}

bool Route::_check(std::string* errorMessage) {
    //include attributes needed
    ElementManager* elements = _parentModel->getElements();
    std::vector<std::string> neededNames = {"Entity.TotalTransferTime", "Entity.Station"};
    std::string neededName;
    for (unsigned int i = 0; i < neededNames.size(); i++) {
        neededName = neededNames[i];
        if (elements->getElement(Util::TypeOf<Attribute>(), neededName) == nullptr) {
            Attribute* attr1 = new Attribute(_parentModel, neededName);
            elements->insert(attr1);
        }
    }
    // include StatisticsCollector needed in EntityType
    std::list<ModelElement*>* enttypes = elements->getElementList(Util::TypeOf<EntityType>())->list();
    for (ModelElement* element : *enttypes) {
        if (element->isReportStatistics())
            static_cast<EntityType*> (element)->addGetStatisticsCollector(element->getName() + ".TransferTime"); // force create this CStat before simulation starts
    }
    bool resultAll = true;
    resultAll &= _parentModel->checkExpression(_routeTimeExpression, "Route time expression", errorMessage);
    if (this->_routeDestinationType == Route::DestinationType::Station) {
        resultAll &= _parentModel->getElements()->check(Util::TypeOf<Station>(), _station, "Station", errorMessage);
        if (resultAll) {
            resultAll &= _station->getEnterIntoStationComponent() != nullptr;
            if (!resultAll) {
                errorMessage->append("Station has no component to enter into it");
            }
        }
    }
    //_model->getParent()->getPluginManager()->
    return resultAll;
}

PluginInformation* Route::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Route>(), &Route::LoadInstance);
    info->setSendTransfer(true);
    info->insertDynamicLibFileDependence("station.so");
    return info;
```

```
}

void Route::_createInternalElements() {
    if (_reportStatistics) {
        if (_numberIn == nullptr) {
            _numberIn = new Counter(_parentModel, getName() + "." + "CountNumberIn", this);
            _childrenElements->insert({"CountNumberIn", _numberIn});
        }
    } else
        if (_numberIn != nullptr) {
        _removeChildrenElements();
    }
}
```

8.491 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Route.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
#include "../elements/Station.h"
#include "../../kernel/simulator/Counter.h"
```

Include dependency graph for Route.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Route

8.492 Route.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Route.h
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:15
00012  */
00013
00014 #ifndef ROUTE_H
00015 #define ROUTE_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018 #include "../elements/Station.h"
00019 #include "../../kernel/simulator/Counter.h"
00020
00053 class Route : public ModelComponent {
00054 public:
00055
00056     enum class DestinationType : int {
00057         Station = 0, BySequence = 1
00058     };
00059 public:
00060     Route(Model* model, std::string name = "");
00061     virtual ~Route() = default;
00062 public:
00063     virtual std::string show();
00064 public:
00065     static PluginInformation* GetPluginInformation();
00066     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00067 public:
00068     void setStation(Station* _station);
00069     Station* getStation() const;
00070     void setRouteTimeExpression(std::string _routeTimeExpression);
00071     std::string getRouteTimeExpression() const;
00072     void setRouteTimeTimeUnit(Util::TimeUnit _routeTimeTimeUnit);
00073     Util::TimeUnit getRouteTimeTimeUnit() const;
00074     void setRouteDestinationType(DestinationType _routeDestinationType);
00075     Route::DestinationType getRouteDestinationType() const;
00076 public:
00077 protected:
00078     virtual void _execute(Entity* entity);
00079     virtual void _initBetweenReplications();
00080     virtual bool _loadInstance(std::map<std::string, std::string>* fields);
00081     virtual std::map<std::string, std::string>* _saveInstance();
00082     virtual bool _check(std::string* errorMessage);
00083     virtual void _createInternalElements();
00084 private:
00085
00086     const struct DEFAULT_VALUES {
00087         std::string routeTimeExpression = "0.0";
00088         Util::TimeUnit routeTimeTimeUnit = Util::TimeUnit::second;
00089         Route::DestinationType routeDestinationType = DestinationType::Station;
00090     } DEFAULT;
00091     std::string _routeTimeExpression = DEFAULT.routeTimeExpression;
00092     Util::TimeUnit _routeTimeTimeUnit = DEFAULT.routeTimeTimeUnit;
00093     Route::DestinationType _routeDestinationType = DEFAULT.routeDestinationType;
00094 private: // association
00095     Station* _station;
00096 private: // children elements
00097     Counter* _numberIn = nullptr;
00098 };
00099
00100 #endif /* ROUTE_H */
00101
```

8.493 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Search.cpp File Reference

```
#include "Search.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Search.cpp:

8.494 Search.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Search.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:20
 */

#include "Search.h"
#include "../../kernel/simulator/Model.h"

Search::Search(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Search>(), name) {
}

std::string Search::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Search::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Search* newComponent = new Search(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Search::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Search::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Search::_initBetweenReplications() {
}

std::map<std::string, std::string>* Search::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Search::_check(std::string* errorMessage) {
```

```
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Search::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Search>(), &Search::LoadInstance);
    // ...
    return info;
}
```

8.495 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Search.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Search.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Search

8.496 Search.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Search.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:20
 */

#ifndef SEARCH_H
#define SEARCH_H

#include "../../kernel/simulator/ModelComponent.h"

class Search : public ModelComponent {
public: // constructors
    Search(Model* model, std::string name = "");
    virtual ~Search() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* SEARCH_H */

```

8.497 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/SeizableItem.cpp File Reference

```
#include "SeizableItem.h"
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/Model.h"
#include <cassert>
```

Include dependency graph for SeizableItem.cpp:

8.498 SeizableItem.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SeizableItem.cpp
 * Author: rlcancian
 *
 * Created on 10 de abril de 2021, 08:45
 */

#include "SeizableItem.h"
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/Model.h"
#include <cassert>

SeizableItem::SeizableItem(ModelElement* resourceOrSet, std::string quantityExpression, SeizableItem::SelectionRule selectionRule, std::string saveAttribute, std::string index) {
    SeizableItem::SeizableType resourceType;
    if (dynamic_cast<Resource*> (resourceOrSet) != nullptr) {
        resourceType = SeizableItem::SeizableType::RESOURCE;
    } else if (dynamic_cast<Set*> (resourceOrSet) != nullptr) {
        resourceType = SeizableItem::SeizableType::SET;
    } else {
        assert(false);
    }
    _seizableType = resourceType;
    _resourceOrSet = resourceOrSet;
    _quantityExpression = quantityExpression;
    _selectionRule = selectionRule;
    _saveAttribute = saveAttribute;
    _index = index;
}

/*
SeizableItem::SeizableItem(ModelElement* resourceOrSet, SeizableItem::SeizableType resourceType, std::string quantityExpression, SeizableItem::SelectionRule selectionRule, std::string saveAttribute, std::string index) {
    // TODO: It could infer _seizableType from resourceOrSet and avoid resourceType parameter
    _seizableType = resourceType;
    _resourceOrSet = resourceOrSet;
    _quantityExpression = quantityExpression;
    _selectionRule = selectionRule;
    _saveAttribute = saveAttribute;
    _index = index;
}
 */

bool SeizableItem::loadInstance(std::map<std::string, std::string>* fields) {
    bool res = true;
    try {
        _seizableType = static_cast<SeizableItem::SeizableType> (LoadField(fields, "seizableType", static_cast<int> (DEFAULT.seizableType)));
        _seizableName = LoadField(fields, "seizable", "");
        _quantityExpression = LoadField(fields, "quantityExpression", DEFAULT.quantityExpression);
        _selectionRule = static_cast<SeizableItem::SelectionRule> (LoadField(fields, "selectionRule", static_cast<int> (DEFAULT.selectionRule)));
        _saveAttribute = LoadField(fields, "saveAttribute", DEFAULT.saveAttribute);
        _index = LoadField(fields, "index", DEFAULT.index);
        if (_elementManager != nullptr) {
            if (_seizableType == SeizableItem::SeizableType::RESOURCE) {
                _resourceOrSet = _elementManager->getElement(Util::TypeOf<Resource>(), _seizableName);
            } else if (_seizableType == SeizableItem::SeizableType::SET) {
                _resourceOrSet = _elementManager->getElement(Util::TypeOf<Set>(), _seizableName);
            }
            assert(_resourceOrSet != nullptr);
        }
    } catch (...) {
        res = false;
    }
    return res;
}

bool SeizableItem::loadInstance(std::map<std::string, std::string>* fields, unsigned int parentIndex) {
    bool res = true;
    std::string num = std::to_string(parentIndex);
    try {
        _seizableType = static_cast<SeizableItem::SeizableType> (LoadField(fields, "seizableType" + num, static_cast<int> (DEFAULT.seizableType)));
        _seizableName = LoadField(fields, "seizable" + num, "");
        _quantityExpression = LoadField(fields, "quantityExpression" + num, DEFAULT.quantityExpression);
```

```
        _selectionRule = static_cast<SeizableItem::SelectionRule> (LoadField(fields, "selectionRule" + num, static_cast<int> (DEFAULT.selectionRule)));
        _saveAttribute = LoadField(fields, "saveAttribute" + num, DEFAULT.saveAttribute);
        _index = LoadField(fields, "index" + num, DEFAULT.index);
        if (_elementManager != nullptr) {
            if (_seizableType == SeizableItem::SeizableType::RESOURCE) {
                _resourceOrSet = _elementManager->getElement(Util::TypeOf<Resource>(), _seizableName);
            } else if (_seizableType == SeizableItem::SeizableType::SET) {
                _resourceOrSet = _elementManager->getElement(Util::TypeOf<Set>(), _seizableName);
            }
            assert(_resourceOrSet != nullptr);
        }
    } catch (...) {
        res = false;
    }
    return res;
}

std::map<std::string, std::string>* SeizableItem::saveInstance(unsigned int parentIndex) {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    std::string num = std::to_string(parentIndex);
    SaveField(fields, "seizableType" + num, static_cast<int> (_seizableType), static_cast<int> (DEFAULT.seizableType));
    //SaveField(fields, "resourceId" + num, _resourceOrSet->getId());
    SaveField(fields, "seizable" + num, _resourceOrSet->getName(), "");
    SaveField(fields, "quantityExpression" + num, _quantityExpression, DEFAULT.quantityExpression);
    SaveField(fields, "selectionRule" + num, static_cast<int> (_selectionRule), static_cast<int> (DEFAULT.selectionRule));
    SaveField(fields, "saveAttribute" + num, _saveAttribute, DEFAULT.saveAttribute);
    SaveField(fields, "index" + num, _index, DEFAULT.index);
    return fields;
}

std::map<std::string, std::string>* SeizableItem::saveInstance() {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
    SaveField(fields, "seizableType", static_cast<int> (_seizableType), static_cast<int> (DEFAULT.seizableType));
    SaveField(fields, "resourceId", _resourceOrSet->getId());
    SaveField(fields, "seizable", _resourceOrSet->getName(), "");
    SaveField(fields, "quantityExpression", _quantityExpression, DEFAULT.quantityExpression);
    SaveField(fields, "selectionRule", static_cast<int> (_selectionRule), static_cast<int> (DEFAULT.selectionRule));
    SaveField(fields, "saveAttribute", _saveAttribute, DEFAULT.saveAttribute);
    SaveField(fields, "index", _index, DEFAULT.index);
    return fields;
}

std::string SeizableItem::show() {
    return "resourceType=" + std::to_string(static_cast<int> (_seizableType)) + ",resource=\"" + _resourceOrSet->getName() + "\",quantityExpression=\"" + _quantityExpression + "\", selectionRule=" + std::to_string(static_cast<int> (_selectionRule)) + ", _saveAttribute=\"" + _saveAttribute + "\",index=\"" + _index + "\"";
}

void SeizableItem::setIndex(std::string index) {
    this->_index = index;
}

std::string SeizableItem::getIndex() const {
    return _index;
}

void SeizableItem::setSaveAttribute(std::string saveAttribute) {
    this->_saveAttribute = saveAttribute;
}

std::string SeizableItem::getSaveAttribute() const {
    return _saveAttribute;
}

void SeizableItem::setSelectionRule(SeizableItem::SelectionRule selectionRule) {
    this->_selectionRule = selectionRule;
}

SeizableItem::SelectionRule SeizableItem::getSelectionRule() const {
    return _selectionRule;
}

void SeizableItem::setQuantityExpression(std::string quantityExpression) {
    this->_quantityExpression = quantityExpression;
}

std::string SeizableItem::getQuantityExpression() const {
    return _quantityExpression;
}

std::string SeizableItem::getResourceName() const {
```

```
    return _seizableName;
}

void SeizableItem::setResource(Resource* resource) {
    this->_resourceOrSet = resource;
    _seizableName = resource->getName();
}

Resource* SeizableItem::getResource() const {
    return static_cast<Resource*> (_resourceOrSet);
}

void SeizableItem::setSet(Set* set) {
    this->_resourceOrSet = set;
    _seizableName = set->getName();
}

Set* SeizableItem::getSet() const {
    return static_cast<Set*> (_resourceOrSet);
}

void SeizableItem::setSeizableType(SeizableItem::SeizableType resourceType) {
    this->_seizableType = resourceType;
}

SeizableItem::SeizableType SeizableItem::getSeizableType() const {
    return _seizableType;
}

void SeizableItem::setLastMemberSeized(unsigned int lastMemberSeized) {
    this->_lastMemberSeized = lastMemberSeized;
}

unsigned int SeizableItem::getLastMemberSeized() const {
    return _lastMemberSeized;
}

ModelElement* SeizableItem::getSeizable() const {
    return _resourceOrSet;
}

void SeizableItem::setElementManager(ElementManager* _elementManager) {
    this->_elementManager = _elementManager;
}

//void SeizableItem::setComponentManager(ComponentManager* _componentManager) {
//  this->_componentManager = _componentManager;
//}

```

8.499 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/SeizableItem.h File Reference

```
#include "../elements/Resource.h"
#include "../elements/Set.h"
#include "../../kernel/simulator/ElementManager.h"
```

Include dependency graph for SeizableItem.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SeizableItem

8.500 SeizableItem.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   SeizableItem.h
 * Author: rlcancian
 *
 * Created on 10 de abril de 2021, 08:45
 */

#ifndef SEIZABLEITEM_H
#define SEIZABLEITEM_H

#include "../elements/Resource.h"
#include "../elements/Set.h"
#include "../../kernel/simulator/ElementManager.h"

// \todo should inhere from a common base to QueueableItem

class SeizableItem {
public:

    enum class SelectionRule : int {
        CYCLICAL = 1, RANDOM = 2, SPECIFICMEMBER = 3, LARGESTREMAININGCAPACITY = 4, SMALLESTNUMBERBUSY = 5
    };

    enum class SeizableType : int {
        RESOURCE = 1, SET = 2
```

```
    };

public:
    SeizableItem(ModelElement* resourceOrSet, std::string quantityExpression = "1",
 SeizableItem::SelectionRule selectionRule = SeizableItem::SelectionRule::LARGESTREMAININGCAPACITY,
 std::string saveAttribute = "", std::string index = "0");

public:
    bool loadInstance(std::map<std::string, std::string>* fields, unsigned int parentIndex);
    std::map<std::string, std::string>* saveInstance(unsigned int parentIndex);
    bool loadInstance(std::map<std::string, std::string>* fields);
    std::map<std::string, std::string>* saveInstance();
public:
    std::string show();
    void setIndex(std::string index);
    std::string getIndex() const;
    void setSaveAttribute(std::string saveAttribute);
    std::string getSaveAttribute() const;
    void setSelectionRule(SelectionRule selectionRule);
    SelectionRule getSelectionRule() const;
    void setQuantityExpression(std::string quantityExpression);
    std::string getQuantityExpression() const;
    std::string getResourceName() const;
    void setResource(Resource* resource);
    Resource* getResource() const;
    void setSet(Set* set);
    Set* getSet() const;
    void setSeizableType(SeizableType resourceType);
    SeizableType getSeizableType() const;
    void setLastMemberSeized(unsigned int lastMemberSeized);
    unsigned int getLastMemberSeized() const;
    ModelElement* getSeizable() const;
    void setElementManager(ElementManager* _elementManager);
    //void setComponentManager(ComponentManager* _componentManager);

private:

    const struct DEFAULT_VALUES {
        const std::string quantityExpression = "1";
        const SeizableItem::SeizableType seizableType = SeizableItem::SeizableType::RESOURCE;
        const SeizableItem::SelectionRule selectionRule =
 SeizableItem::SelectionRule::LARGESTREMAININGCAPACITY;
        const std::string saveAttribute = "";
        const std::string index = "0";
    } DEFAULT;

    SeizableType _seizableType = DEFAULT.seizableType;
    SelectionRule _selectionRule = DEFAULT.selectionRule;
    std::string _saveAttribute = DEFAULT.saveAttribute;
    std::string _index = DEFAULT.index;
    ModelElement* _resourceOrSet;
    std::string _seizableName;
    std::string _quantityExpression;
    unsigned int _lastMemberSeized = 0;
private:
    //ComponentManager* _componentManager;
    ElementManager* _elementManager;
};

#endif /* SEIZABLEITEM_H */
```

8.501 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Seize.cpp File Reference

```
#include "Seize.h"
#include "../elements/Resource.h"
#include "../../kernel/simulator/Attribute.h"
#include <assert.h>
#include <cmath>
```

Include dependency graph for Seize.cpp:

8.502 Seize.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Seize.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Agosto de 2018, 16:17
 */

#include "Seize.h"
#include "../elements/Resource.h"
#include "../../kernel/simulator/Attribute.h"
#include <assert.h>
#include <cmath>

Seize::Seize(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Seize>(), name) {
}

std::string Seize::show() {
    std::string txt = ModelComponent::show() +
            "priority=" + std::to_string(_priority) +
            "seizeRequests={";
    unsigned short i = 0;
    for (SeizableItem* item : *_seizeRequests->list()) {
        txt += "request" + std::to_string(i++) + "=[" + item->show() + "],";
    }
    txt = txt.substr(0, txt.length() - 1) + "}";
    return txt;
}

void Seize::setPriority(unsigned short _priority) {
    this->_priority = _priority;
}

unsigned short Seize::getPriority() const {
    return _priority;
}

void Seize::setAllocationType(unsigned int _allocationType) {
    this->_allocationType = _allocationType;
}

unsigned int Seize::getAllocationType() const {
    return _allocationType;
}

/*
void Seize::setQueueName(std::string queueName) throw () {
        Queue* queue = dynamic_cast<Queue*>
 (_parentModel->getElements()->getElement(Util::TypeOf<Queue>(), queueName));
        if (queue != nullptr) {
                _queue = queue;
        } else {
                throw std::invalid_argument("Queue does not exist");
        }
```

```
}

//void Seize::setResourceName(std::string resourceName) throw () {
//  Resource* resource = dynamic_cast<Resource*>(_parentModel->elements()->element(Util::TypeOf<Resource>(), resourceName));
//  if (resource != nullptr) {
//      _resource = resource;
//  } else {
//      throw std::invalid_argument("Resource does not exist");
//  }
//}

//std::string Seize::getResourceName() const {
//  return _seizeRequest->resource()->name();
//}

std::string Seize::getQueueName() const {
        return _queue->getName();
}

//void Seize::setResource(Resource* resource) {
//  this->_resource = resource;
 * TODO: move the handler to the list of SeizableItens
//  _resource->addReleaseResourceEventHandler(Resource::SetResourceEventHandler<Seize>(&Seize:://_handlerForResourceEvent, this));
//}

//Resource* Seize::getResource() const {
//  return _resource;
//}

void Seize::setQueue(Queue* queue) {
        this->_queue = queue;
}

Queue* Seize::getQueue() const {
        return _queue;
}
 */

void Seize::setQueue(Queue* queue) {
    setQueueableItem(new QueueableItem(queue));
}

Queue* Seize::_getQueue() const {
    if (_queueableItem->getQueueableType() == QueueableItem::QueueableType::QUEUE) {
        return _queueableItem->getQueue();
    } else {
        //assume it is a SET
        unsigned int index = _parentModel->parseExpression(_queueableItem->getIndex());
        return static_cast<Queue*> (_queueableItem->getSet()->getElementSet()->getAtRank(index));
    }
}

void Seize::_handlerForResourceEvent(Resource* resource) { //TODO Resource is useless now
    Queue* queue = _getQueue();
    Waiting* first = queue->first();
    if (first != nullptr) { // there are entities waiting in the queue
        // check if all Seize requests can be seized by the entity in the queue
        bool canSeizeAll = true;
        unsigned int quantityRequested, quantityAvailable;
        for (SeizableItem* seizable : *_seizeRequests->list()) {
            Resource* resource = _getResourceFromSeizableItem(seizable, first->getEntity());
            quantityRequested = _parentModel->parseExpression(seizable->getQuantityExpression());
            quantityAvailable = resource->getCapacity() - resource->getNumberBusy();
            if (quantityAvailable < quantityRequested) {
                canSeizeAll = false;
                break;
            }
        }
        if (canSeizeAll) {
            queue->removeElement(first);
            //_parentModel->getTracer()->traceSimulation(tnow, first->getEntity(), this, "Waiting entity " + first->getEntity()->getName() + " removed from queue and will try to seize the resources");// now seizes " + std::to_string(quantity) + " elements of resource \"" + resource->getName() + "\"");
            _parentModel->getTracer()->trace("Waiting entity " + first->getEntity()->getName() + " removed from queue and will try to seize the resources"); // now seizes " + std::to_string(quantity) + " elements of resource \"" + resource->getName() + "\"");
            _parentModel->sendEntityToComponent(first->getEntity(), this); // move waiting entity from queue to this component
        }
        /*
        if (request->getResource() == resource) {
            unsigned int quantity = _parentModel->parseExpression(request->getQuantityExpression());
            if ((resource->getCapacity() - resource->getNumberBusy()) >= quantity) { //enought
```

```
quantity to seize
                double tnow = _parentModel->getSimulation()->getSimulatedTime();
                resource->seize(quantity, tnow);
                _parentModel->getFutureEvents()->insert(new Event(tnow, first->getEntity(), this->getNextComponents()->getFrontConnection()));
                queue->removeElement(first);
                _parentModel->getTracer()->traceSimulation(tnow, first->getEntity(), this, "Waiting entity " + first->getEntity()->getName() + " removed from queue and now seizes " + std::to_string(quantity) + " elements of resource \"" + resource->getName() + "\"");
            }
        }
         */
    }
}

List<SeizableItem*>* Seize::getSeizeRequests() const {
    return _seizeRequests;
}

void Seize::setQueueableItem(QueueableItem* _queueableItem) {
    this->_queueableItem = _queueableItem;
}

QueueableItem* Seize::getQueueableItem() const {
    return _queueableItem;
}

void Seize::setSaveAttribute(std::string _saveAttribute) {
    this->_saveAttribute = _saveAttribute;
}

std::string Seize::getSaveAttribute() const {
    return _saveAttribute;
}

Resource* Seize::_getResourceFromSeizableItem(SeizableItem* seizable, Entity* entity) {
    Resource* resource;
    if (seizable->getSeizableType() == SeizableItem::SeizableType::RESOURCE) {
        resource = seizable->getResource();
    } else { // assume SET
        SeizableItem::SelectionRule rule = seizable->getSelectionRule();
        Set* set = seizable->getSet();
        unsigned int index = 0;
        switch (rule) {
            case SeizableItem::SelectionRule::CYCLICAL:
                index = (seizable->getLastMemberSeized() + 1) % _seizeRequests->list()->size();
                break;
            case SeizableItem::SelectionRule::LARGESTREMAININGCAPACITY:
                // \todo
                break;
            case SeizableItem::SelectionRule::RANDOM:
                index = std::trunc(rand() * _seizeRequests->list()->size()); // TODO: RANDOM IS REALLY A PROBLEM!! NOW IT MAY CAUSE AN ERROR (DEQUEUE AN ENTITY BECAUSE IT CAN SEIZE ALL REQUESTS, BUT ANOTHER RANDOM REQUEST MY BE SELECTED AFTER, IT IT MAY BE BUSY
                break;
            case SeizableItem::SelectionRule::SMALLESTNUMBERBUSY:
                // \todo
                break;
            case SeizableItem::SelectionRule::SPECIFICMEMBER:
                index = _parentModel->parseExpression(seizable->getIndex());
                break;
        }
        if (_saveAttribute != "") {
            entity->setAttributeValue(_saveAttribute, index);
        }
        _parentModel->getTracer()->trace("Member of set " + set->getName() + " chosen index " + std::to_string(index), Util::TraceLevel::L8_detailed);
        resource = static_cast<Resource*> (set->getElementSet()->getAtRank(index));
        assert(resource != nullptr);
    }
    return resource;
}

void Seize::_execute(Entity* entity) {
    for (SeizableItem* seizable : *_seizeRequests->list()) {
        Resource* resource = _getResourceFromSeizableItem(seizable, entity);
        unsigned int quantity = _parentModel->parseExpression(seizable->getQuantityExpression());
        if (resource->getCapacity() - resource->getNumberBusy() < quantity) { // not enought free quantity to allocate. Entity goes to the queue
            WaitingResource* waitingRec = new WaitingResource(entity, _parentModel->getSimulation()->getSimulatedTime(), quantity, this);
            Queue* queue;
            if (_queueableItem->getQueueableType() == QueueableItem::QueueableType::QUEUE) {
                queue = _queueableItem->getQueue();
            } else { // assume SET
                Set* set = _queueableItem->getSet();
                unsigned int index = _parentModel->parseExpression(_queueableItem->getIndex());
```

```
                _parentModel->getTracer()->trace("Member of set " + set->getName() + " chosen index " + std::to_string(index), Util::TraceLevel::L8_detailed);
                queue = static_cast<Queue*> (set->getElementSet()->getAtRank(index));
            }
            queue->insertElement(waitingRec); // ->list()->insert(waitingRec);

    _parentModel->getTracer()->traceSimulation(_parentModel->getSimulation()->getSimulatedTime(), entity, this, "Entity starts to wait for resource in queue \"" + queue->getName() + "\" with " + std::to_string(queue->size()) + " elements");
            return;
        } else { // alocate the resource
            resource->seize(quantity, _parentModel->getSimulation()->getSimulatedTime());

    _parentModel->getTracer()->traceSimulation(_parentModel->getSimulation()->getSimulatedTime(), entity, this, entity->getName() + " seizes " + std::to_string(quantity) + " elements of resource \"" + resource->getName() + "\" (capacity:" + std::to_string(resource->getCapacity()) + ", numberbusy:" + std::to_string(resource->getNumberBusy()) + ")");
        }
    }
    _parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

void Seize::_initBetweenReplications() {
    //ModelElement::InitBetweenReplications(_queueableItem->getQueueable());
    //for (std::list<SeizableItem*>::iterator it = _seizeRequests->list()->begin(); it != _seizeRequests->list()->end(); it++) {
    //  (*it)->setLastMemberSeized(0);
    //  ModelElement::InitBetweenReplications((*it)->getSeizable());
    //}
}

bool Seize::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_allocationType = LoadField(fields, "allocationType", DEFAULT.allocationType);
        this->_priority = LoadField(fields, "priority", DEFAULT.priority);
        this->_saveAttribute = LoadField(fields, "saveAttribute", DEFAULT.saveAttribute);
        //std::string queueName = LoadField(fields, "queue", "");
        //Queue* queue = dynamic_cast<Queue*> (_parentModel->getElements()->getElement(Util::TypeOf<Queue>(), queueName));
        //this->_queue = queue;
        //QueueableItem
        //QueueableItem*
        _queueableItem = new QueueableItem(nullptr);
        _queueableItem->setElementManager(_parentModel->getElements());
        _queueableItem->loadInstance(fields);
        unsigned short numRequests = LoadField(fields, "resquestSize", DEFAULT.seizeRequestSize);
        for (unsigned short i = 0; i < numRequests; i++) {
            SeizableItem* Item = new SeizableItem(nullptr);
            Item->setElementManager(_parentModel->getElements());
            Item->loadInstance(fields, i);
        }

    }
    return res;
}

std::map<std::string, std::string>* Seize::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance(); //Util::TypeOf<Seize>());
    SaveField(fields, "allocationType", _allocationType, DEFAULT.allocationType);
    SaveField(fields, "priority=", _priority, DEFAULT.priority);
    SaveField(fields, "saveAttribute=", _saveAttribute, DEFAULT.saveAttribute);
    //SaveField(fields, "queueId", _queue->getId());
    //SaveField(fields, "queue", _queue->getName());
    std::map<std::string, std::string>* queueablefields = _queueableItem->saveInstance();
    fields->insert(queueablefields->begin(), queueablefields->end());
    SaveField(fields, "resquestSize", _seizeRequests->size(), DEFAULT.seizeRequestSize);
    unsigned short i = 0;
    for (SeizableItem* request : *_seizeRequests->list()) {
        std::map<std::string, std::string>* seizablefields = request->saveInstance(i);
        fields->insert(seizablefields->begin(), seizablefields->end());
        i++;
    }
    return fields;
}

bool Seize::_check(std::string* errorMessage) {
    bool resultAll = true;
    for (SeizableItem* seizable : *_seizeRequests->list()) {
        resultAll &= _parentModel->checkExpression(seizable->getQuantityExpression(), "quantity", errorMessage);
        if (seizable->getSeizableType() == SeizableItem::SeizableType::RESOURCE) {
            resultAll &= _parentModel->getElements()->check(Util::TypeOf<Resource>(), seizable->getResource(), "Resource", errorMessage);

    seizable->getResource()->addReleaseResourceEventHandler(Resource::SetResourceEventHandler<Seize>(&Seize::_handlerForRes
```

```
this), this, _priority);
        } else if (seizable->getSeizableType() == SeizableItem::SeizableType::SET) {
            resultAll &= _parentModel->getElements()->check(Util::TypeOf<Set>(), seizable->getSet(), "Set", errorMessage);
            Resource* rec;
            for (ModelElement* elem : *seizable->getSet()->getElementSet()->list()) {
                rec = static_cast<Resource*> (elem);

rec->addReleaseResourceEventHandler(Resource::SetResourceEventHandler<Seize>(&Seize::_handlerForResourceEvent, this), this, _priority);
            }
        }

        // \todo: Should be checking saveAttribute, index, etc
    }
    // \todo Check QueueableItem
    if (_queueableItem->getQueueableType() == QueueableItem::QueueableType::QUEUE) {
        resultAll &= _parentModel->getElements()->check(Util::TypeOf<Queue>(), _queueableItem->getQueue(), "Queueable Queue", errorMessage);
    } else if (_queueableItem->getQueueableType() == QueueableItem::QueueableType::SET) {
        resultAll &= _parentModel->getElements()->check(Util::TypeOf<Set>(), _queueableItem->getSet(), "Queueable Set", errorMessage);
    }

    if (_saveAttribute != "") { // check if saveAttribute is an attribute
        resultAll &= _parentModel->getElements()->check(Util::TypeOf<Attribute>(), _saveAttribute, "Save Attribute", true, errorMessage);
    }
    //resultAll &= _parentModel->getElements()->check(Util::TypeOf<Attribute>(), _saveAttribute, "SaveAttribute", false, errorMessage);
    return resultAll;
}

PluginInformation* Seize::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Seize>(), &Seize::LoadInstance);
    info->insertDynamicLibFileDependence("queue.so");
    info->insertDynamicLibFileDependence("resource.so");
    return info;
}

ModelComponent* Seize::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Seize* newComponent = new Seize(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;

}

```

8.503 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Seize.h File Reference

```
#include <string>
#include "../../kernel/simulator/ModelComponent.h"
#include "../../kernel/simulator/Model.h"
#include "../elements/Resource.h"
#include "../elements/Queue.h"
#include "../../kernel/simulator/Plugin.h"
#include "SeizableItem.h"
#include "QueueableItem.h"
```

Include dependency graph for Seize.h:

This graph shows which files directly or indirectly include this file:

Classes

- class WaitingResource
- class Seize

8.504 Seize.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Seize.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Agosto de 2018, 16:17
00012  */
00013
00014 #ifndef SEIZE_H
00015 #define SEIZE_H
00016
00017 #include <string>
00018 #include "../../kernel/simulator/ModelComponent.h"
00019 #include "../../kernel/simulator/Model.h"
00020 #include "../elements/Resource.h"
00021 #include "../elements/Queue.h"
00022 #include "../../kernel/simulator/Plugin.h"
00023 #include "SeizableItem.h"
00024 #include "QueueableItem.h"
00025
00026 class WaitingResource : public Waiting {
00027 public:
00028     WaitingResource(Entity* entity, double timeStartedWaiting, unsigned int quantity, ModelComponent*
       component) : Waiting(entity, timeStartedWaiting, component) {
00029         _quantity = quantity;
00030     }
00031     WaitingResource(const WaitingResource& orig) : Waiting(orig) {
00032     }
00033
00034     virtual ~WaitingResource() = default;
00035 public:
00036     virtual std::string show() {
00037         return Waiting::show() +
```

```
                ",quantity=" + std::to_string(this->_quantity);
    }
public:
    unsigned int getQuantity() const {
        return _quantity;
    }
private:
    unsigned int _quantity;
};

class Seize : public ModelComponent {
public:
    Seize(Model* model, std::string name = "");
    virtual ~Seize() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public: // get & set
    void setPriority(unsigned short _priority);
    unsigned short getPriority() const;
    void setAllocationType(unsigned int _allocationType);
    unsigned int getAllocationType() const;
    // indirect access to Queue* and Resource*
    //void setResourceName(std::string _resourceName) throw ();
    //std::string getResourceName() const;
    //void setQueueName(std::string queueName) throw ();
    //std::string getQueueName() const;
    void setQueue(Queue* queue);
    //Queue* getQueue() const;
    List<SeizableItem*>* getSeizeRequests() const;
    void setQueueableItem(QueueableItem* _queueableItem);
    QueueableItem* getQueueableItem() const;
    void setSaveAttribute(std::string _saveAttribute);
    std::string getSaveAttribute() const;
protected:
    virtual void _execute(Entity* entity);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual void _initBetweenReplications();
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    //virtual void _createInternalElements();
private:
    void _handlerForResourceEvent(Resource* resource);
    Resource* _getResourceFromSeizableItem(SeizableItem* seizable, Entity* entity);
    Queue* _getQueue() const;
private:

    const struct DEFAULT_VALUES {
        unsigned int allocationType = 0; // uint ? enum?
        unsigned short priority = 0;
        unsigned int seizeRequestSize = 1;
        std::string saveAttribute = "";
    } DEFAULT;
    unsigned int _allocationType = DEFAULT.allocationType; // uint ? enum?
    unsigned short _priority = DEFAULT.priority;
    std::string _saveAttribute = "";
    QueueableItem* _queueableItem; //Queue* _queue; // usually has a queue, but not always (it could be a hold or a set)
    List<SeizableItem*>* _seizeRequests = new List<SeizableItem*>();
private: // not gets or sets
    //  unsigned int _lastMemberSeized = 0; //  now _seizeRequest is a list and it was moved to SeizableItem
};

#endif /* SEIZE_H */

```

8.505 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Separate.cpp File Reference

```
#include "Separate.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Separate.cpp:

8.506 Separate.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Separate.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:14
00012  */
00013
00014 #include "Separate.h"
00015
00016 #include "../../kernel/simulator/Model.h"
00017
00018 Separate::Separate(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Separate>(),
       name) {
00019 }
00020
00021 std::string Separate::show() {
00022     return ModelComponent::show() + "";
00023 }
00024
00025 ModelComponent* Separate::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
00026     Separate* newComponent = new Separate(model);
00027     try {
00028         newComponent->_loadInstance(fields);
00029     } catch (const std::exception& e) {
00030
00031     }
00032     return newComponent;
00033 }
00034
00035 void Separate::_execute(Entity* entity) {
00036     //Entity* cloned = _parentModel->createEntity(entity, false);  // false? check
00037     this->_parentModel->sendEntityToComponent(entity, getConnections()->getFrontConnection());
00038     //this->_parentModel->sendEntityToComponent(cloned, nextComponents()->getConnectionAtRank(1),
       0.0);
00039 }
00040
00041 bool Separate::_loadInstance(std::map<std::string, std::string>* fields) {
00042     bool res = ModelComponent::_loadInstance(fields);
00043     if (res) {
00044         // \todo: not implemented yet
00045     }
00046     return res;
00047 }
00048
00049 void Separate::_initBetweenReplications() {
00050 }
00051
00052 std::map<std::string, std::string>* Separate::_saveInstance() {
00053     std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
00054     // \todo: not implemented yet
```

```
    return fields;
}

bool Separate::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Separate::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Separate>(),
 &Separate::LoadInstance);
    // ...
    return info;
}

```

8.507 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Separate.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Separate.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Separate

8.508 Separate.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Separate.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:14
 */

#ifndef SEPARATE_H
#define SEPARATE_H

#include "../../kernel/simulator/ModelComponent.h"

class Separate : public ModelComponent {
public: // constructors
    Separate(Model* model, std::string name = "");
    virtual ~Separate() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* SEPARATE_H */

```

8.509 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Signal.cpp File Reference

```
#include "Signal.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Signal.cpp:

8.510 Signal.cpp

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Signal.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:20
 */

#include "Signal.h"

#include "../../kernel/simulator/Model.h"

Signal::Signal(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Signal>(), name) {
}

std::string Signal::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Signal::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Signal* newComponent = new Signal(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Signal::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Signal::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Signal::_initBetweenReplications() {
}

std::map<std::string, std::string>* Signal::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Signal::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Signal::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Signal>(), &Signal::LoadInstance);
    // ...
    return info;
}
```

8.511 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/components/Signal.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Signal.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Signal

8.512 Signal.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Signal.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:20
 */

#ifndef SIGNAL_H
#define SIGNAL_H

#include "../../kernel/simulator/ModelComponent.h"

class Signal : public ModelComponent {
public: // constructors
    Signal(Model* model, std::string name = "");
    virtual ~Signal() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* SIGNAL_H */

```

8.513 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Start.cpp File Reference

```
#include "Start.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Start.cpp:

8.514 Start.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Start.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:15
 */

#include "Start.h"

#include "../../kernel/simulator/Model.h"

Start::Start(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Start>(), name) {
}

std::string Start::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Start::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Start* newComponent = new Start(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Start::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Start::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}
```

```
00047
00048 void Start::_initBetweenReplications() {
00049 }
00050
00051 std::map<std::string, std::string>* Start::_saveInstance() {
00052     std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
00053     // \todo: not implemented yet
00054     return fields;
00055 }
00056
00057 bool Start::_check(std::string* errorMessage) {
00058     bool resultAll = true;
00059     // \todo: not implemented yet
00060     *errorMessage += "";
00061     return resultAll;
00062 }
00063
00064 PluginInformation* Start::GetPluginInformation() {
00065     PluginInformation* info = new PluginInformation(Util::TypeOf<Start>(), &Start::LoadInstance);
00066     // ...
00067     return info;
00068 }
00069
00070
```

8.515 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Start.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Start.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Start

8.516 Start.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Start.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:15
 */

#ifndef START_H
#define START_H

#include "../../kernel/simulator/ModelComponent.h"

class Start : public ModelComponent {
public: // constructors
    Start(Model* model, std::string name = "");
    virtual ~Start() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* START_H */

```

8.517 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Stop.cpp File Reference

```
#include "Stop.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Stop.cpp:

8.518 Stop.cpp

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Stop.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:15
 */

#include "Stop.h"

#include "../../kernel/simulator/Model.h"

Stop::Stop(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Stop>(), name) {
}

std::string Stop::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Stop::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Stop* newComponent = new Stop(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Stop::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Stop::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Stop::_initBetweenReplications() {
}

std::map<std::string, std::string>* Stop::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Stop::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Stop::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Stop>(), &Stop::LoadInstance);
    // ...
    return info;
}
```

8.519 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Stop.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Stop.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Stop

8.520 Stop.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Stop.h
00009  * Author: rlcancian
00010  *
00011  * Created on 11 de Setembro de 2019, 13:15
00012  */
00013
00014 #ifndef STOP_H
00015 #define STOP_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018
00036 class Stop : public ModelComponent {
00037 public: // constructors
00038     Stop(Model* model, std::string name = "");
00039     virtual ~Stop() = default;
00040 public: // virtual
00041     virtual std::string show();
00042 public: // static
00043     static PluginInformation* GetPluginInformation();
00044     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00045 protected: // virtual
00046     virtual void _execute(Entity* entity);
00047     virtual void _initBetweenReplications();
```

```
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* STOP_H */

```

8.521 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Store.cpp File Reference

```
#include "Store.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Store.cpp:

8.522 Store.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Store.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:07
 */

#include "Store.h"
#include "../../kernel/simulator/Model.h"

Store::Store(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Store>(), name) {
}

std::string Store::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Store::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Store* newComponent = new Store(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
```

```
}

void Store::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Store::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Store::_initBetweenReplications() {
}

std::map<std::string, std::string>* Store::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Store::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Store::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Store>(), &Store::LoadInstance);
    // ...
    return info;
}
```

8.523 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Store.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Store.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Store

8.524 Store.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Store.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:07
 */

#ifndef STORE_H
#define STORE_H

#include "../../kernel/simulator/ModelComponent.h"

class Store : public ModelComponent {
public: // constructors
    Store(Model* model, std::string name = "");
    virtual ~Store() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* STORE_H */

```

8.525 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Submodel.cpp File Reference

```
#include "Submodel.h"
```

`#include "../../kernel/simulator/Model.h"`

Include dependency graph for Submodel.cpp:

8.526 Submodel.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Submodel.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:04
 */

#include "Submodel.h"

#include "../../kernel/simulator/Model.h"

Submodel::Submodel(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Submodel>(), name) {
}

std::string Submodel::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Submodel::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Submodel* newComponent = new Submodel(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Submodel::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Submodel::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Submodel::_initBetweenReplications() {
}

std::map<std::string, std::string>* Submodel::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
```

```
}

bool Submodel::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Submodel::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Submodel>(),
 &Submodel::LoadInstance);
    // ...
    return info;
}
```

8.527 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Submodel.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Submodel.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Submodel

8.528 Submodel.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Submodel.h
00009  * Author: rlcancian
00010  *
00011  * Created on 11 de Setembro de 2019, 13:04
00012  */
00013
00014 #ifndef SUBMODEL_H
00015 #define SUBMODEL_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018
00022 class Submodel : public ModelComponent {
00023 public: // constructors
00024     Submodel(Model* model, std::string name = "");
00025     virtual ~Submodel() = default;
00026 public: // virtual
00027     virtual std::string show();
00028 public: // static
00029     static PluginInformation* GetPluginInformation();
00030     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00031 protected: // virtual
00032     virtual void _execute(Entity* entity);
00033     virtual void _initBetweenReplications();
00034     virtual bool _loadInstance(std::map<std::string, std::string>* fields);
00035     virtual std::map<std::string, std::string>* _saveInstance();
00036     virtual bool _check(std::string* errorMessage);
00037 private: // methods
00038 private: // attributes 1:1
00039 private: // attributes 1:n
00040 };
00041
00042
00043 #endif /* SUBMODEL_H */
00044
```

8.529 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Unstore.cpp File Reference

```
#include "Unstore.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Unstore.cpp:

8.530 Unstore.cpp

```
00001 /*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Unstore.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:08
 */

#include "Unstore.h"
#include "../../kernel/simulator/Model.h"

Unstore::Unstore(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Unstore>(), name) {
}

std::string Unstore::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Unstore::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Unstore* newComponent = new Unstore(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newComponent;
}

void Unstore::_execute(Entity* entity) {
    _parentModel->getTracer()->trace("I'm just a dummy model and I'll just send the entity forward");
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

bool Unstore::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        // \todo: not implemented yet
    }
    return res;
}

void Unstore::_initBetweenReplications() {
}

std::map<std::string, std::string>* Unstore::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    // \todo: not implemented yet
    return fields;
}

bool Unstore::_check(std::string* errorMessage) {
    bool resultAll = true;
    // \todo: not implemented yet
    *errorMessage += "";
    return resultAll;
}

PluginInformation* Unstore::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Unstore>(), &Unstore::LoadInstance);
    // ...
    return info;
}
```

8.531 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Unstore.h File Reference

```
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Unstore.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Unstore

8.532 Unstore.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Unstore.h
00009  * Author: rlcancian
00010  *
00011  * Created on 11 de Setembro de 2019, 13:08
00012  */
00013
00014 #ifndef UNSTORE_H
00015 #define UNSTORE_H
00016
00017 #include "../../kernel/simulator/ModelComponent.h"
00018
00045 class Unstore : public ModelComponent {
00046 public: // constructors
00047     Unstore(Model* model, std::string name = "");
00048     virtual ~Unstore() = default;
00049 public: // virtual
00050     virtual std::string show();
00051 public: // static
00052     static PluginInformation* GetPluginInformation();
00053     static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00054 protected: // virtual
```

```
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
private: // attributes 1:1
private: // attributes 1:n
};


#endif /* UNSTORE_H */

```

8.533 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Write.cpp File Reference

```
#include "Write.h"
#include "../../kernel/simulator/Model.h"
#include <fstream>
```

Include dependency graph for Write.cpp:

8.534 Write.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Write.cpp
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:06
 */

#include "Write.h"
#include "../../kernel/simulator/Model.h"

#include <fstream>

Write::Write(Model* model, std::string name) : ModelComponent(model, Util::TypeOf<Write>(), name) {
}

std::string Write::show() {
    return ModelComponent::show() + "";
}

ModelComponent* Write::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Write* newComponent = new Write(model);
    try {
        newComponent->_loadInstance(fields);
    } catch (const std::exception& e) {
```

```

    }
    return newComponent;
}

List<WriteText*>* Write::writeElements() const {
    return _writeElements;
}

void Write::setFilename(std::string _filename) {
    this->_filename = _filename;
}

std::string Write::filename() const {
    return _filename;
}

void Write::setWriteToType(WriteToType _writeToType) {
    this->_writeToType = _writeToType;
}

Write::WriteToType Write::writeToType() const {
    return _writeToType;
}

void Write::_execute(Entity* entity) {
    std::ofstream savefile;
    if (this->_writeToType == Write::WriteToType::FILE) {
        savefile.open(_filename, std::ofstream::app);
    }
    std::string message = "";
    for (WriteText* msgElem : *_writeElements->list()) {
        if (msgElem->isExpression) {
            message += std::to_string(_parentModel->parseExpression(msgElem->text));
        } else {
            message += msgElem->text;
        }
        if (msgElem->newline) {
            if (this->_writeToType == Write::WriteToType::SCREEN) { //\todo: Write To FILE not implemented
                _parentModel->getTracer()->trace(Util::TraceLevel::L2_results, message);
            } else if (this->_writeToType == Write::WriteToType::FILE) {
                // open file
                savefile « message « std::endl;
            }
            message = "";
        }
    }
    if (this->_writeToType == Write::WriteToType::FILE) {
        savefile.close();
    }
    this->_parentModel->sendEntityToComponent(entity, this->getConnections()->getFrontConnection());
}

void Write::_initBetweenReplications() {
    try {
        std::ofstream savefile;
        savefile.open(_filename, std::ofstream::app);
        savefile « "# Replication number " « _parentModel->getSimulation()->getCurrentReplicationNumber() « "/" « _parentModel->getSimulation()->getNumberOfReplications() « std::endl;
        savefile.close();
    } catch (...) {

    }
}

bool Write::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelComponent::_loadInstance(fields);
    if (res) {
        this->_writeToType = static_cast<WriteToType>(std::stoi((*fields->find("writeToType")).second));
        unsigned short writesSize = std::stoi((*fields->find("writesSize")).second);
        for (unsigned short i = 0; i < writesSize; i++) {
            std::string text = (*fields->find(text)).second;
            bool isExpression = static_cast<bool> (std::stoi((*fields->find("isExpression")).second));
            bool newline = static_cast<bool> (std::stoi((*fields->find("newline")).second));
            if (isExpression) {
                if (newline) {
                    this->_writeElements->insert(new WritelnExpression(text));
                } else {
                    this->_writeElements->insert(new WriteExpression(text));
                }
            } else {
                if (newline) {
                    this->_writeElements->insert(new WritelnText(text));
                } else {
```

```
                    this->_writeElements->insert(new WriteText(text));
                }
            }
        }
    }
    return res;
}

std::map<std::string, std::string>* Write::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelComponent::_saveInstance();
    SaveField(fields, "writeToType", static_cast<int> (_writeToType));
    SaveField(fields, "writesSize", _writeElements->size(), 0u);
    unsigned short i = 0;
    WriteText* writeElem;
    for (std::list<WriteText*>::iterator it = _writeElements->list()->begin(); it != _writeElements->list()->end(); it++, i++) {
        writeElem = (*it);
        SaveField(fields, "isExpression" + std::to_string(i), writeElem->isExpression, writeElem->DEFAULT.isExpression);
        SaveField(fields, "newline" + std::to_string(i), writeElem->newline, writeElem->DEFAULT.newline);
        SaveField(fields, "text" + std::to_string(i), writeElem->text, writeElem->DEFAULT.text);
    }
    //this->_writeElements
    return fields;
}

bool Write::_check(std::string* errorMessage) {
    bool resultAll = true;
    WriteText* msgElem;
    unsigned short i = 0;
    std::list<WriteText*>* msgs = this->_writeElements->list();
    for (std::list<WriteText*>::iterator it = msgs->begin(); it != msgs->end(); it++) {
        msgElem = (*it);
        i++;
        if (msgElem->isExpression) {
            resultAll &= _parentModel->checkExpression(msgElem->text, "writeExpression" + std::to_string(i), errorMessage);
        }
    }
    // when cheking the model (before simulating it), remove the file if exists
    std::remove(_filename.c_str());

    return resultAll;
}

PluginInformation* Write::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Write>(), &Write::LoadInstance);
    // ...
    return info;
}
```

8.535 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/components/Write.h File Reference

`#include "../../kernel/simulator/ModelComponent.h"`

Include dependency graph for Write.h:

This graph shows which files directly or indirectly include this file:

Classes

- class WriteText
- struct WriteText::DEFAULT_VALUES
- class WriteInText
- struct WriteInText::DEFAULT_VALUES
- class WriteExpression
- struct WriteExpression::DEFAULT_VALUES
- class WriteInExpression
- struct WriteInExpression::DEFAULT_VALUES
- class Write

8.536 Write.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Write.h
 * Author: rlcancian
 *
 * Created on 11 de Setembro de 2019, 13:06
 */

#ifndef WRITE_H
#define WRITE_H

#include "../../kernel/simulator/ModelComponent.h"

class WriteText {
public:
    WriteText(std::string text = "") {
        this->text = text;
        this->isExpression = false;
        this->newline = false;
    }

    const struct DEFAULT_VALUES {
        const std::string text = "";
        const bool isExpression = false;
        const bool newline = false;
    } DEFAULT;
    std::string text = DEFAULT.text;
    bool isExpression = DEFAULT.isExpression;
    bool newline = DEFAULT.newline;
};

class WriteInText : public WriteText {
public:
    WriteInText(std::string text = "") : WriteText(text) {
        this->newline = true;
    }

    const struct DEFAULT_VALUES {
        const std::string text = "";
        const bool isExpression = false;
        const bool newline = true;
    } DEFAULT;
```

```
};

class WriteExpression : public WriteText {
public:
    WriteExpression(std::string text = "") : WriteText(text) {
        this->isExpression = true;
    }

    const struct DEFAULT_VALUES {
        const std::string text = "";
        const bool isExpression = true;
        const bool newline = false;
    } DEFAULT;
};

class WritelnExpression : public WriteExpression {
public:
    WritelnExpression(std::string text = "") : WriteExpression(text) {
        this->isExpression = true;
        this->newline = true;
    }

    const struct DEFAULT_VALUES {
        const std::string text = "";
        const bool isExpression = true;
        const bool newline = true;
    } DEFAULT;
};

class Write : public ModelComponent {
public:

    enum class WriteToType : int {
        SCREEN = 1, FILE = 2
    };



public: // constructors
    Write(Model* model, std::string name = "");
    virtual ~Write() = default;
public: // virtual
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelComponent* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    List<WriteText*>* writeElements() const;
    void setFilename(std::string _filename);
    std::string filename() const;
    void setWriteToType(WriteToType _writeToType);
    Write::WriteToType writeToType() const;

protected: // virtual
    virtual void _execute(Entity* entity);
    virtual void _initBetweenReplications();
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private: // methods
    //std::string _buildText();
private: // attributes 1:1

    const struct DEFAULT_VALUES {
        WriteToType writeToType = Write::WriteToType::SCREEN;
        std::string filename = "";
    } DEFAULT;
    WriteToType _writeToType = DEFAULT.writeToType;
    std::string _filename = DEFAULT.filename;
private: // attributes 1:n
    List<WriteText*>* _writeElements = new List<WriteText*>();
};


#endif /* WRITE_H */

```

8.537 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/EntityGroup.cpp File Reference

```
#include "EntityGroup.h"
```

```
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
```

Include dependency graph for EntityGroup.cpp:

8.538 EntityGroup.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Group.cpp
 * Author: rlcancian
 *
 * Created on 12 de Junho de 2019, 19:00
 */

#include "EntityGroup.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"

EntityGroup::EntityGroup(Model* model, std::string name) : ModelElement(model,
 Util::TypeOf<EntityGroup>(), name) {
}

EntityGroup::~EntityGroup() {
    //_parentModel->elements()->remove(Util::TypeOf<StatisticsCollector>(), _cstatNumberInGroup);
}

std::string EntityGroup::show() {
    return ModelElement::show() +
            ",entities=" + this->_list->show();
}

void EntityGroup::insertElement(Entity* element) {
    _list->insert(element);
    this->_cstatNumberInGroup->getStatistics()->getCollector()->addValue(_list->size());
}

void EntityGroup::removeElement(Entity* element) {
    //double tnow = this->_elements->getParentModel()->simulation()->getSimulatedTime();
    _list->remove(element);
    this->_cstatNumberInGroup->getStatistics()->getCollector()->addValue(_list->size());
}

void EntityGroup::initBetweenReplications() {
    this->_list->clear();
    this->_cstatNumberInGroup->getStatistics()->getCollector()->clear();
}

unsigned int EntityGroup::size() {
    return _list->size();
}

Entity* EntityGroup::first() {
    return _list->front();
}

//List<Waiting*>* Group::getList() const {
//  return _list;
```

```
//}

PluginInformation* EntityGroup::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<EntityGroup>(), &EntityGroup::LoadInstance);
    return info;
}

ModelElement* EntityGroup::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    EntityGroup* newElement = new EntityGroup(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool EntityGroup::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* EntityGroup::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance(); //Util::TypeOf<Group>());
    return fields;
}

bool EntityGroup::_check(std::string* errorMessage) {
    std::string newNeededAttributeName = "Entity.Group";
    if (_parentModel->getElements()->getElement(Util::TypeOf<Attribute>(), newNeededAttributeName) == nullptr) {
        new Attribute(_parentModel, newNeededAttributeName);
    }
    *errorMessage += "";
    return true;
}

void EntityGroup::_createInternalElements() {
    if (_reportStatistics) {
        if (_cstatNumberInGroup == nullptr) {
            _cstatNumberInGroup = new StatisticsCollector(_parentModel, "NumberInGroup", this);
            _childrenElements->insert({"NumberInGroup", _cstatNumberInGroup});
        }
    } else
        if (_cstatNumberInGroup != nullptr) {
        _removeChildrenElements();
    }
}


```

8.539 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/EntityGroup.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/Entity.h"
#include <map>
#include <list>
```

Include dependency graph for EntityGroup.h:

This graph shows which files directly or indirectly include this file:

Classes

- class EntityGroup

8.540 EntityGroup.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Group.h
 * Author: rlcancian
 *
 * Created on 12 de Junho de 2019, 19:00
 */

#ifndef ENTITYGROUP_H
#define ENTITYGROUP_H

#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/Entity.h"
#include <map>
#include <list>

/*
 The EnetityGroup represents a set of entities grouped togheter somehow. The entities in this group
 are kept into a internal queue. Usually one entity representing all grouped entities is generated by
 ModelComponents and the EntityGroup it represented is accessed through an attribute.
 */
class EntityGroup : public ModelElement {
public:
    EntityGroup(Model* model, std::string name = "");
    virtual ~EntityGroup();
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    void insertElement(Entity* element);
```

```
    void removeElement(Entity* element);
    unsigned int size();
    Entity* first();
public:
    void initBetweenReplications();
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private:
    void _initCStats();
private: //1::n
    List<Entity*>* _list = new List<Entity*>();
private: // child inner element
    StatisticsCollector* _cstatNumberInGroup = nullptr;
};

#endif /* ENTITYGROUP_H */

```

8.541 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Failure.cpp File Reference

`#include "Failure.h"`

Include dependency graph for Failure.cpp:

8.542 Failure.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Failure.cpp
 * Author: rlcancian
 *
 * Created on 20 de Failureembro de 2019, 20:07
 */

#include "Failure.h"

Failure::Failure(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Failure>(), name)
 {
}

std::string Failure::show() {
    return ModelElement::show() +
            "";
}

PluginInformation* Failure::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Failure>(), &Failure::LoadInstance);
    return info;
}

```

```
ModelElement* Failure::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Failure* newElement = new Failure(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool Failure::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
            //this->_attributeName = (*fields->find("attributeName")).second;
            //this->_orderRule = static_cast<OrderRule>
 (std::stoi((*fields->find("orderRule")).second));
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* Failure::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<Failure>());
    //SaveField(fields, "orderRule", std::to_string(static_cast<int> (this->_orderRule)));
    //SaveField(fields, "attributeName", "\""+this->_attributeName+"\"");
    return fields;
}

bool Failure::_check(std::string* errorMessage) {
    bool resultAll = true;
    // resultAll |= ...
    *errorMessage += "";
    return resultAll;
}

ParserChangesInformation* Failure::_getParserChangesInformation() {
    ParserChangesInformation* changes = new ParserChangesInformation();
    //changes->getProductionToAdd()->insert(...);
    //changes->getTokensToAdd()->insert(...);
    return changes;
}
```

8.543 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Failure.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/PluginInformation.h"
```

Include dependency graph for Failure.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Failure

8.544 Failure.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Failure.h
 * Author: rlcancian
 *
 * Created on 20 de Failureembro de 2019, 20:07
 */

#ifndef FAILURE_H
#define FAILURE_H


#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
//#include "ParserChangesInformation.h"
#include "../../kernel/simulator/PluginInformation.h"

class Failure : public ModelElement {
public:
    Failure(Model* model, std::string name = "");
    virtual ~Failure() = default;
public: // static
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
    static PluginInformation* GetPluginInformation();
public:
    virtual std::string show();

protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
protected: // could be overriden by derived classes
    virtual bool _check(std::string* errorMessage);
    virtual ParserChangesInformation* _getParserChangesInformation();

};

#endif /* FAILURE_H */

```

8.545 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/File.cpp File Reference

```
#include "File.h"
```

Include dependency graph for File.cpp:

8.546 File.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   File.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 20 de Fileembro de 2019, 20:07
00012  */
00013
00014 #include "File.h"
00015
00016 File::File(Model* model, std::string name) : ModelElement(model, Util::TypeOf<File>(), name) {
00017     //_elems = elems;
00018 }
00019
00020 std::string File::show() {
00021     return ModelElement::show() +
00022             "";
00023 }
00024
00025 PluginInformation* File::GetPluginInformation() {
00026     PluginInformation* info = new PluginInformation(Util::TypeOf<File>(), &File::LoadInstance);
00027     return info;
00028 }
00029
00030 ModelElement* File::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
00031     File* newElement = new File(model);
00032     try {
00033         newElement->_loadInstance(fields);
00034     } catch (const std::exception& e) {
00035
00036     }
00037     return newElement;
00038 }
00039
00040 bool File::_loadInstance(std::map<std::string, std::string>* fields) {
00041     bool res = ModelElement::_loadInstance(fields);
00042     if (res) {
00043         try {
00044             //this->_attributeName = (*fields->find("attributeName")).second;
00045             //this->_orderRule = static_cast<OrderRule>
       (std::stoi((*fields->find("orderRule")).second));
00046         } catch (...) {
00047         }
00048     }
00049     return res;
00050 }
00051
00052 std::map<std::string, std::string>* File::_saveInstance() {
00053     std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
       //Util::TypeOf<File>());
00054     //SaveField(fields, "orderRule", std::to_string(static_cast<int> (this->_orderRule)));
00055     //SaveField(fields, "attributeName", "\""+this->_attributeName+"\"");
00056     return fields;
00057 }
00058
00059 bool File::_check(std::string* errorMessage) {
00060     bool resultAll = true;
```

```
    // resultAll |= ...
    *errorMessage += "";
    return resultAll;
}

ParserChangesInformation* File::_getParserChangesInformation() {
    ParserChangesInformation* changes = new ParserChangesInformation();
    //changes->getProductionToAdd()->insert(...);
    //changes->getTokensToAdd()->insert(...);
    return changes;
}

```

8.547 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/File.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/PluginInformation.h"
```

Include dependency graph for File.h:

This graph shows which files directly or indirectly include this file:

Classes

- class File

8.548 File.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
/*
 * File:   File.h
 * Author: rlcancian
 *
 * Created on 20 de Fileembro de 2019, 20:07
 */

#ifndef FILE_H
#define FILE_H


#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
//#include "ParserChangesInformation.h"
#include "../../kernel/simulator/PluginInformation.h"

class File : public ModelElement {
public:
    File(Model* model, std::string name = "");
    virtual ~File() = default;
public: // static
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
    static PluginInformation* GetPluginInformation();
public:
    virtual std::string show();

protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
protected: // could be overriden by derived classes
    virtual bool _check(std::string* errorMessage);
    virtual ParserChangesInformation* _getParserChangesInformation();

};

#endif /* FILE_H */

```

8.549 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Queue.cpp File Reference

```
#include "Queue.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
```

Include dependency graph for Queue.cpp:

8.550 Queue.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
/*
 * File:   Queue.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Agosto de 2018, 17:12
 */

#include "Queue.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"

Queue::Queue(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Queue>(), name) {
}

Queue::~Queue() {
    //_parentModel->elements()->remove(Util::TypeOf<StatisticsCollector>(), _cstatNumberInQueue);
    //_parentModel->elements()->remove(Util::TypeOf<StatisticsCollector>(), _cstatTimeInQueue);
}

std::string Queue::show() {
    return ModelElement::show() +
            ",waiting=" + this->_list->show();
}

void Queue::insertElement(Waiting* element) {
    _list->insert(element);
    if (_reportStatistics)
        this->_cstatNumberInQueue->getStatistics()->getCollector()->addValue(_list->size());
}

void Queue::removeElement(Waiting* element) {
    double tnow = _parentModel->getSimulation()->getSimulatedTime();
    _list->remove(element);
    if (_reportStatistics) {
        this->_cstatNumberInQueue->getStatistics()->getCollector()->addValue(_list->size());
        double timeInQueue = tnow - element->getTimeStartedWaiting();
        this->_cstatTimeInQueue->getStatistics()->getCollector()->addValue(timeInQueue);
    }
}

void Queue::initBetweenReplications() {
    this->_list->clear();
}

unsigned int Queue::size() {
    return _list->size();
}

Waiting* Queue::first() {
    return _list->front();
}

Waiting* Queue::getAtRank(unsigned int rank) {
    return _list->getAtRank(rank);
}

void Queue::setAttributeName(std::string _attributeName) {
    this->_attributeName = _attributeName;
}

std::string Queue::getAttributeName() const {
    return _attributeName;
}

void Queue::setOrderRule(OrderRule _orderRule) {
    this->_orderRule = _orderRule;
}

Queue::OrderRule Queue::getOrderRule() const {
    return _orderRule;
}

double Queue::sumAttributesFromWaiting(Util::identification attributeID) {
    double sum = 0.0;
    for (Waiting* waiting : *_list->list()) {
        sum += waiting->getEntity()->getAttributeValue(attributeID);
    }
    return sum;
}

double Queue::getAttributeFromWaitingRank(unsigned int rank, Util::identification attributeID) {
    Waiting* wait = _list->getAtRank(rank);
    if (wait != nullptr) {
        return wait->getEntity()->getAttributeValue(attributeID);
    }
    return 0.0;
}
```

```
PluginInformation* Queue::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Queue>(), &Queue::LoadInstance);
    return info;
}

ModelElement* Queue::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Queue* newElement = new Queue(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool Queue::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
            this->_attributeName = LoadField(fields, "attributeName", DEFAULT.attributeName);
            this->_orderRule = static_cast<OrderRule> (LoadField(fields, "orderRule", static_cast<int>(DEFAULT.orderRule)));
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* Queue::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance(); //Util::TypeOf<Queue>());
    SaveField(fields, "orderRule", static_cast<int> (this->_orderRule), static_cast<int>(DEFAULT.orderRule));
    SaveField(fields, "attributeName", this->_attributeName, DEFAULT.attributeName);
    return fields;
}

bool Queue::_check(std::string* errorMessage) {
    return _parentModel->getElements()->check(Util::TypeOf<Attribute>(), _attributeName, "AttributeName", false, errorMessage);
}

void Queue::_createInternalElements() {
    if (_reportStatistics) {
        if (_cstatNumberInQueue == nullptr) {
            _cstatNumberInQueue = new StatisticsCollector(_parentModel, getName() + "." + "NumberInQueue", this); /* \todo: ++ WHY THIS INSERT "DISPOSE" AND "10ENTITYTYPE" STATCOLL ?? */
            _cstatTimeInQueue = new StatisticsCollector(_parentModel, getName() + "." + "TimeInQueue", this);
            _childrenElements->insert({"NumberInQueue", _cstatNumberInQueue});
            _childrenElements->insert({"TimeInQueue", _cstatTimeInQueue});
        }
    } else if (_cstatNumberInQueue != nullptr) {
        // \todo: remove
        _removeChildrenElements();
        //_childrenElements->remove(_cstatNumberInQueue);
        //_childrenElements->remove(_cstatTimeInQueue);
        //_cstatNumberInQueue->~StatisticsCollector();
        //_cstatTimeInQueue->~StatisticsCollector();
    }
}

ParserChangesInformation * Queue::_getParserChangesInformation() {
    ParserChangesInformation* changes = new ParserChangesInformation();
    //changes->getProductionToAdd()->insert(...);
    //changes->getTokensToAdd()->insert(...);
    return changes;
}
```

8.551 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Queue.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/util/List.h"
#include "../../kernel/simulator/Entity.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/StatisticsCollector.h"
```

```
#include "../../kernel/simulator/Plugin.h"
#include "../../kernel/simulator/ModelComponent.h"
```

Include dependency graph for Queue.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Waiting
- class Queue

8.552 Queue.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Queue.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 21 de Agosto de 2018, 17:12
00012  */
00013
00014 #ifndef QUEUE_H
00015 #define QUEUE_H
00016
00017 #include "../../kernel/simulator/ModelElement.h"
00018 #include "../../kernel/util/List.h"
00019 #include "../../kernel/simulator/Entity.h"
00020 #include "../../kernel/simulator/ElementManager.h"
00021 #include "../../kernel/simulator/StatisticsCollector.h"
00022 #include "../../kernel/simulator/Plugin.h"
00023 #include "../../kernel/simulator/ModelComponent.h"
00024
00025 class Waiting {
00026 public:
```

```
    Waiting(Entity* entity, double timeStartedWaiting, ModelComponent* component, unsigned int componentOutputNumber = 0) {
        _entity = entity;
        _component = component;
        _timeStartedWaiting = timeStartedWaiting;
        _componentOutputNumber = componentOutputNumber;
    }

    virtual ~Waiting() = default;
public:
    virtual std::string show() {
        return //ModelElement::show()+
        ",entity=" + std::to_string(_entity->getId()) +
                ",component=\"" + _component->getName() + "\"" +
                ",timeStatedWaiting=" + std::to_string(_timeStartedWaiting);
    }
public:
    double getTimeStartedWaiting() const {
        return _timeStartedWaiting;
    }
    ModelComponent* getComponent() const {
        return _component;
    }
    Entity* getEntity() const {
        return _entity;
    }
    unsigned int getComponentOutputNumber() const {
        return _componentOutputNumber;
    }
private:
    Entity* _entity;
    ModelComponent* _component;
    double _timeStartedWaiting;
    unsigned int _componentOutputNumber;
};

class Queue : public ModelElement {
public:

    enum class OrderRule : int {
        FIFO = 1, LIFO = 2, HIGHESTVALUE = 3, SMALLESTVALUE = 4
    };

public:
    Queue(Model* model, std::string name = "");
    virtual ~Queue();
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    void insertElement(Waiting* element);
    void removeElement(Waiting* element);
    unsigned int size();
    Waiting* first();
    Waiting* getAtRank(unsigned int rank);
    void setAttributeName(std::string _attributeName);
    std::string getAttributeName() const;
    void setOrderRule(OrderRule _orderRule);
    Queue::OrderRule getOrderRule() const;
public: // to implement SIMAN functions
    double sumAttributesFromWaiting(Util::identification attributeID); // use to implement SIMAN SAQUE function
    double getAttributeFromWaitingRank(unsigned int rank, Util::identification attributeID);
public:
    void initBetweenReplications();
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
    virtual ParserChangesInformation* _getParserChangesInformation();

private:
    void _initCStats();
private: //1::n
    List<Waiting*>* _list = new List<Waiting*>();
private: //1::1

    const struct DEFAULT_VALUES {
        OrderRule orderRule = OrderRule::FIFO;
        std::string attributeName = "";
    } DEFAULT;
    OrderRule _orderRule = DEFAULT.orderRule;
    std::string _attributeName = DEFAULT.attributeName;
private: // inner children elements
```

```
    StatisticsCollector* _cstatNumberInQueue = nullptr;
    StatisticsCollector* _cstatTimeInQueue;
};

#endif /* QUEUE_H */

```

8.553 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Resource.cpp File Reference

```
#include "Resource.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Resource.cpp:

8.554 Resource.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Resource.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Agosto de 2018, 16:52
 */

#include "Resource.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Model.h"

Resource::Resource(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Resource>(), name) {
    _resourceEventHandlers->setSortFunc([](const SortedResourceEventHandler* a, const SortedResourceEventHandler * b) {
        return a->second < b->second;
    });

    GetterMember getter = DefineGetterMember<Resource>(this, &Resource::getCapacity);
    SetterMember setter = DefineSetterMember<Resource>(this, &Resource::setCapacity);
    model->getControls()->insert(new SimulationControl(Util::TypeOf<Resource>(), getName() + ".Capacity", getter, setter));

    GetterMember getter2 = DefineGetterMember<Resource>(this, &Resource::getCostPerUse);
    SetterMember setter2 = DefineSetterMember<Resource>(this, &Resource::setCostPerUse);
    model->getControls()->insert(new SimulationControl(Util::TypeOf<Resource>(), getName() + ".CostPerUse", getter2, setter2));
    // ...
}

std::string Resource::show() {
    return ModelElement::show() +
```

```
            ",capacity=" + strTruncIfInt(std::to_string(_capacity)) +
            ",costBusyByour=" + strTruncIfInt(std::to_string(_costBusyHour)) +
            ",costIdleByour=" + strTruncIfInt(std::to_string(_costIdleHour)) +
            ",costPerUse=" + strTruncIfInt(std::to_string(_costPerUse)) +
            ",state=" + strTruncIfInt(std::to_string(static_cast<int> (_resourceState)));
}

void Resource::seize(unsigned int quantity, double tnow) {
    _numberBusy += quantity;
    if (_reportStatistics)
        _numSeizes->incCountValue(quantity);
    _lastTimeSeized = tnow;
    _resourceState = Resource::ResourceState::BUSY;
    // \todo implement costs
}

void Resource::release(unsigned int quantity, double tnow) {
    if (_numberBusy >= quantity) {
        _numberBusy -= quantity;
    } else {
        _numberBusy = 0;
    }
    if (_numberBusy == 0) {
        _resourceState = Resource::ResourceState::IDLE;
    }
    double timeSeized = tnow - _lastTimeSeized;
    if (_reportStatistics) {
        _numReleases->incCountValue(quantity);
        _cstatTimeSeized->getStatistics()->getCollector()->addValue(timeSeized);
        _totalTimeSeized->incCountValue(timeSeized);
    }
    //
    _lastTimeSeized = timeSeized;
    _notifyReleaseEventHandlers();
}

void Resource::_initBetweenReplications() {
    this->_lastTimeSeized = 0.0;
    this->_numberBusy = 0;
    if (_reportStatistics) {
        this->_numSeizes->clear();
        this->_numReleases->clear();
    }
}

void Resource::setResourceState(ResourceState _resourceState) {
    this->_resourceState = _resourceState;
}

Resource::ResourceState Resource::getResourceState() const {
    return _resourceState;
}

void Resource::setCapacity(unsigned int _capacity) {
    this->_capacity = _capacity;
}

unsigned int Resource::getCapacity() const {
    return _capacity;
}

void Resource::setCostBusyHour(double _costBusyHour) {
    this->_costBusyHour = _costBusyHour;
}

double Resource::getCostBusyHour() const {
    return _costBusyHour;
}

void Resource::setCostIdleHour(double _costIdleHour) {
    this->_costIdleHour = _costIdleHour;
}

double Resource::getCostIdleHour() const {
    return _costIdleHour;
}

void Resource::setCostPerUse(double _costPerUse) {
    this->_costPerUse = _costPerUse;
}

double Resource::getCostPerUse() const {
    return _costPerUse;
}

unsigned int Resource::getNumberBusy() const {
    return _numberBusy;
```

```
}

void Resource::addReleaseResourceEventHandler(ResourceEventHandler eventHandler, ModelComponent* component, unsigned int priority) {
    //if (_resourceEventHandlers->find(eventHandler) != _resourceEventHandlers->list()->end()) {
    ModelComponent* compHandler;
    for (SortedResourceEventHandler* sreh : *_resourceEventHandlers->list()) {
        compHandler = sreh->first.second;
        if (compHandler == component) {
            return; // already exists. Do not insert again
        }
    }
    SortedResourceEventHandler* sortedEventHandler = new SortedResourceEventHandler({
        {eventHandler, component}, priority
    });
    _resourceEventHandlers->insert(sortedEventHandler);
}

double Resource::getLastTimeSeized() const {
    return _lastTimeSeized;
}

void Resource::_notifyReleaseEventHandlers() {
    for (SortedResourceEventHandler* sortedHandler : *_resourceEventHandlers->list()) {
        ResourceEventHandler handler = sortedHandler->first.first;
        handler(this);
    }
}

PluginInformation* Resource::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Resource>(), &Resource::LoadInstance);
    return info;
}

ModelElement* Resource::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Resource* newElement = new Resource(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool Resource::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        _capacity = LoadField(fields, "capacity", DEFAULT.capacity);
        _costBusyHour = LoadField(fields, "costBusyHour", DEFAULT.cost);
        _costIdleHour = LoadField(fields, "costIdleHour", DEFAULT.cost);
        _costPerUse = LoadField(fields, "costPerUse", DEFAULT.cost);
        _resourceState = static_cast<Resource::ResourceState> (LoadField(fields, "resourceState", static_cast<int> (DEFAULT.resourceState)));
    }
    return res;
}

std::map<std::string, std::string>* Resource::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance(); //Util::TypeOf<Resource>());
    SaveField(fields, "capacity", _capacity, DEFAULT.capacity);
    SaveField(fields, "costBusyHour", _costBusyHour, DEFAULT.cost);
    SaveField(fields, "costIdleHour", _costIdleHour, DEFAULT.cost);
    SaveField(fields, "costPerUse", _costPerUse, DEFAULT.cost);
    SaveField(fields, "resourceState", static_cast<int> (_resourceState), static_cast<int> (DEFAULT.resourceState));
    return fields;
}

bool Resource::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}

void Resource::_createInternalElements() {
    if (_reportStatistics && _cstatTimeSeized == nullptr) {
        _cstatTimeSeized = new StatisticsCollector(_parentModel, getName() + "." + "TimeSeized", this);
        _totalTimeSeized = new Counter(_parentModel, getName() + "." + "TotalTimeSeized", this);
        _numSeizes = new Counter(_parentModel, getName() + "." + "Seizes", this);
        _numReleases = new Counter(_parentModel, getName() + "." + "Releases", this);
        _childrenElements->insert({"TimeSeized", _cstatTimeSeized});
        _childrenElements->insert({"TotalTimeSeized", _totalTimeSeized});
        _childrenElements->insert({"Seizes", _numSeizes});
        _childrenElements->insert({"Releases", _numReleases});
    } else if (!_reportStatistics && _cstatTimeSeized != nullptr) {
```

```
        _removeChildrenElements();
    }
}
```

8.555 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Resource.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/StatisticsCollector.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Resource.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Resource

8.556 Resource.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Resource.h
 * Author: rafael.luiz.cancian
 *
 * Created on 21 de Agosto de 2018, 16:52
 */

#ifndef RESOURCE_H
#define RESOURCE_H
```

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/StatisticsCollector.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Counter.h"
#include "../../kernel/simulator/Plugin.h"

class SeizableItem;

class Resource : public ModelElement {
public:
    typedef std::function<void(Resource*) > ResourceEventHandler;
    typedef std::pair<std::pair<ResourceEventHandler, ModelComponent*>, unsigned int> SortedResourceEventHandler;
    template<typename Class>
    static ResourceEventHandler SetResourceEventHandler(void (Class::*function)(Resource*), Class * object) {
        return std::bind(function, object, std::placeholders::_1);
    }

    enum class ResourceState : int {
        IDLE = 1, BUSY = 2, FAILED = 3, INACTIVE = 4, OTHER = 5
    };

public:
    //Resource(Model* model);
    Resource(Model* model, std::string name = "");
    virtual ~Resource() = default;
public:
    virtual std::string show();
public:
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    void seize(unsigned int quantity, double tnow);
    void release(unsigned int quantity, double tnow);
public: // g&s
    void setResourceState(ResourceState _resourceState);
    Resource::ResourceState getResourceState() const;
    void setCapacity(unsigned int _capacity);
    unsigned int getCapacity() const;
    void setCostBusyHour(double _costBusyHour);
    double getCostBusyHour() const;
    void setCostIdleHour(double _costIdleHour);
    double getCostIdleHour() const;
    void setCostPerUse(double _costPerUse);
    double getCostPerUse() const;
public: // gets
    unsigned int getNumberBusy() const;
public:
    void addReleaseResourceEventHandler(ResourceEventHandler eventHandler, ModelComponent* component, unsigned int priority);
    double getLastTimeSeized() const;
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
    virtual void _initBetweenReplications();
private:
    void _notifyReleaseEventHandlers();
    //private:
    //    ElementManager* _elems;
private:

    const struct DEFAULT_VALUES {
        unsigned int capacity = 1;
        double cost = 1.0;
        ResourceState resourceState = ResourceState::IDLE;
    } DEFAULT;
    unsigned int _capacity = DEFAULT.capacity;
    double _costBusyHour = DEFAULT.cost;
    double _costIdleHour = DEFAULT.cost;
    double _costPerUse = DEFAULT.cost;
    ResourceState _resourceState = DEFAULT.resourceState;
private: // only gets
    unsigned int _numberBusy = 0;
    //unsigned int _numberOut = 0;
    double _lastTimeSeized = 0.0; // \todo: It won't work for resources with capacity>1, when not all capacity is seized and them some more are seized. Seized time of first units will be lost. I don't have a solution so far
private: // not gets nor sets
    //unsigned int _seizes = 0;
    //double _whenSeized; // same as last? check
private: //1::n
    List<SortedResourceEventHandler*>* _resourceEventHandlers = new List<SortedResourceEventHandler*>();
```

```
    //aFailures:    TStringList;
    //std::list<Failure*>* _failures;
private: // inner children elements
    StatisticsCollector* _cstatTimeSeized = nullptr;
    Counter* _totalTimeSeized;
    Counter* _numSeizes;
    Counter* _numReleases;
};

#endif /* RESOURCE_H */

```

8.557 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Schedule.cpp File Reference

`#include "Schedule.h"`

Include dependency graph for Schedule.cpp:

8.558 Schedule.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Schedule.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:16
 */

#include "Schedule.h"

Schedule::Schedule(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Schedule>(),
     name) {
}

std::string Schedule::show() {
}

PluginInformation* Schedule::GetPluginInformation() {
}

ModelElement* Schedule::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
}

bool Schedule::_loadInstance(std::map<std::string, std::string>* fields) {
}

std::map<std::string, std::string>* Schedule::_saveInstance() {
}

bool Schedule::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}

```

8.559 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Schedule.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/PluginInformation.h"
```

Include dependency graph for Schedule.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Schedule

8.560 Schedule.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Schedule.h
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:16
00012  */
00013
00014 #ifndef SCHEDULE_H
00015 #define SCHEDULE_H
00016
00017 class Model;
00018
00068 #include "../../kernel/simulator/ModelElement.h"
00069 #include "../../kernel/simulator/ElementManager.h"
00070 #include "../../kernel/simulator/PluginInformation.h"
```

```
00071
00072 class Schedule : public ModelElement {
00073 public: // constructors
00074     Schedule(Model* model, std::string name = "");
00075     virtual ~Schedule() = default;
00076 public: // virtual
00077     virtual std::string show();
00078 public: // static
00079     static PluginInformation* GetPluginInformation();
00080     static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00081 public:
00082 protected:
00083     virtual bool _loadInstance(std::map<std::string, std::string>* fields);
00084     virtual std::map<std::string, std::string>* _saveInstance();
00085     virtual bool _check(std::string* errorMessage);
00086 private:
00087 };
00088
00089 #endif /* SCHEDULE_H */
00090
```

8.561 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Sequence.cpp File Reference

```
#include "Sequence.h"
#include "../../kernel/simulator/Attribute.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Sequence.cpp:

8.562 Sequence.cpp

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Sequence.cpp
00009  * Author: rlcancian
00010  *
00011  * Created on 03 de Junho de 2019, 15:12
00012  */
00013
00014 #include "Sequence.h"
00015 #include "../../kernel/simulator/Attribute.h"
00016 #include "../../kernel/simulator/Model.h"
00017
00018 Sequence::Sequence(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Sequence>(),
       name) {
00019 }
00020
00021 std::string Sequence::show() {
```

```
    std::string msg = ModelElement::show();
    return msg;
}

PluginInformation* Sequence::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Sequence>(), &Sequence::LoadInstance);
    return info;
}

ModelElement* Sequence::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Sequence* newElement = new Sequence(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

List<SequenceStep*>* Sequence::getSteps() const {
    return _steps;
}

bool Sequence::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* Sequence::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance(); //Util::TypeOf<Sequence>());
    return fields;
}

bool Sequence::_check(std::string* errorMessage) {
    /* include attributes needed */
    std::vector<std::string> neededNames = {"Entity.Sequence", "Entity.SequenceStep"};
    std::string neededName;
    for (unsigned int i = 0; i < neededNames.size(); i++) {
        neededName = neededNames[i];
        if (_parentModel->getElements()->getElement(Util::TypeOf<Attribute>(), neededName) == nullptr) {
            new Attribute(_parentModel, neededName);
            //_parentModel->insert(attr1);
        }
    }
    *errorMessage += "";
    return true;
}

SequenceStep::SequenceStep(Station* station, std::list<Assignment*>* assignments) {
    this->_station = station;
    if (assignments != nullptr)
        _assignments = assignments;
    else
        _assignments = new std::list<Assignment*>();
}

bool SequenceStep::_loadInstance(std::map<std::string, std::string>* fields, unsigned int parentIndex) {
    bool res = true;
    std::string num = std::to_string(parentIndex);
    std::string destination, expression;
    try {
        std::string stationName = LoadField(fields, "station" + num, "");
        // \todo _station =
        unsigned int assignmentsSize = LoadField(fields, "assignmentSize" + num, DEFAULT.assignmentsSize);
        for (unsigned short i = 0; i < assignmentsSize; i++) {
            destination = LoadField(fields, "destination" + num + "_" + std::to_string(i));
            expression = LoadField(fields, "expression" + num + "_" + std::to_string(i));
            _assignments->insert(_assignments->end(), new Assignment(destination, expression));
        }
    } catch (...) {
        res = false;
    }
    return res;
}

std::map<std::string, std::string>* SequenceStep::_saveInstance(unsigned int parentIndex) {
    std::map<std::string, std::string>* fields = new std::map<std::string, std::string>();
```

```
    std::string num = std::to_string(parentIndex);
    SaveField(fields, "station" + num, _station->getName());
    //SaveField(fields, "resourceId" + num, _resourceOrSet->getId());
    SaveField(fields, "assignmentSize" + num, _assignments->size(), DEFAULT.assignmentsSize);
    unsigned short i = 0;
    for (Assignment* assm : *_assignments) {
        SaveField(fields, "destination" + num + "_" + std::to_string(i), assm->getDestination());
        SaveField(fields, "expression" + num + "_" + std::to_string(i), assm->getExpression());
        i++;
    }
    return fields;
}

bool SequenceStep::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = true;
    try {
    } catch (...) {
        res = false;
    }
    return res;
}

std::map<std::string, std::string>* SequenceStep::_saveInstance() {
    //\TODO
}

std::list<SequenceStep::Assignment*>* SequenceStep::getAssignments() const {
    return _assignments;
}

void SequenceStep::setStation(Station* _station) {
    this->_station = _station;
}

Station* SequenceStep::getStation() const {
    return _station;
}
```

8.563 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Sequence.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/PluginInformation.h"
#include "Station.h"
```

Include dependency graph for Sequence.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SequenceStep
- class SequenceStep::Assignment
- class Sequence

8.564 Sequence.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Sequence.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:12
 */

#ifndef SEQUENCE_H
#define SEQUENCE_H

#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/PluginInformation.h"
#include "Station.h"

class SequenceStep : PersistentObject_base {
public:

    class Assignment {
    public:
        Assignment(std::string destination, std::string expression) {
            this->_destination = destination;
            this->_expression = expression;
            // an assignment is always in the form:
            // (destinationType) destination = expression
        };
        void setDestination(std::string _destination) {
            this->_destination = _destination;
        }
        std::string getDestination() const {
            return _destination;
        }
        void setExpression(std::string _expression) {
            this->_expression = _expression;
        }
        std::string getExpression() const {
            return _expression;
        }
    private:
        std::string _destination = "";
        std::string _expression = "";

    };

public:

    SequenceStep(Station* station, std::list<Assignment*>* assignments = nullptr);
public: // virtual

    virtual bool _loadInstance(std::map<std::string, std::string>* fields, unsigned int parentIndex);
    virtual std::map<std::string, std::string>* _saveInstance(unsigned int parentIndex);
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();

public:

    std::list<SequenceStep::Assignment*>* getAssignments() const;
    void setStation(Station* _station);
    Station* getStation() const;
private:

    const struct DEFAULT_VALUES {
        const unsigned int assignmentsSize = 0;
    } DEFAULT;
    Station* _station;
    std::list<Assignment*>* _assignments;
};

```

```
class Sequence : public ModelElement {
public:


public:
    Sequence(Model* model, std::string name = "");
    virtual ~Sequence() = default;
public:
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
    List<SequenceStep*>* getSteps() const;
public:
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
private:
    List<SequenceStep*>* _steps = new List<SequenceStep*>();
};

#endif /* SEQUENCE_H */

```

8.565 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Set.cpp File Reference

```
#include "Set.h"
#include "../../kernel/simulator/Model.h"
```

Include dependency graph for Set.cpp:

8.566 Set.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Set.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:11
 */

#include "Set.h"
#include "../../kernel/simulator/Model.h"

Set::Set(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Set>(), name) {
}

std::string Set::show() {
```

```
    return ModelElement::show() +
            "";
}

void Set::setSetOfType(std::string _setOfType) {
    this->_setOfType = _setOfType;
}

std::string Set::getSetOfType() const {
    return _setOfType;
}

List<ModelElement*>* Set::getElementSet() const {
    return _elementSet;
}

PluginInformation* Set::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Set>(), &Set::LoadInstance);
    return info;
}

ModelElement* Set::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Set* newElement = new Set(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool Set::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
            _setOfType = LoadField(fields, "type", DEFAULT.setOfType);
            unsigned int memberSize = LoadField(fields, "memberSize", DEFAULT.membersSize);
            for (unsigned int i = 0; i < memberSize; i++) {
                std::string memberName = LoadField(fields, "member" + std::to_string(i));
                ModelElement* member = _parentModel->getElements()->getElement(_setOfType,
 memberName);
                if (member == nullptr) { // not found. That's a problem. Resource not loaded yet (or
 mismatch name
                    _parentModel->getTracer()->trace("ERROR: Could not found " + _setOfType + " set
 member named \"" + memberName + "\"", Util::TraceLevel::Ll_errorFatal);
                } else {//found
                    _elementSet->insert(member);
                }
            }
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* Set::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<Set>());
    SaveField(fields, "type", _setOfType, DEFAULT.setOfType);
    SaveField(fields, "membersSize", _elementSet->size(), DEFAULT.membersSize);
    unsigned int i = 0;
    for (ModelElement* element : *_elementSet->list()) {
        SaveField(fields, "member" + std::to_string(i), element->getName());
        i++;
    }
    return fields;
}

bool Set::_check(std::string* errorMessage) {
    bool resultAll = true;
    if (_elementSet->size() > 0) {
        std::string typeOfFirstElement = _elementSet->front()->getClassname();
        if (_setOfType == "") {
            _setOfType = typeOfFirstElement;
        } else if (_setOfType != typeOfFirstElement) {
            resultAll = false;
            *errorMessage += "Set is of type \"" + _setOfType + "\" and first element is of type \"" +
 typeOfFirstElement + "\"";
        }
    }
    *errorMessage += "";
    return resultAll;
}

ParserChangesInformation* Set::_getParserChangesInformation() {
    ParserChangesInformation* changes = new ParserChangesInformation();
    //changes->getProductionToAdd()->insert(...);
```

```
    //changes->getTokensToAdd()->insert(...);
    return changes;
}
```

8.567 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Set.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/ParserChangesInformation.h"
#include "../../kernel/simulator/PluginInformation.h"
#include "../../kernel/simulator/EntityType.h"
```

Include dependency graph for Set.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Set

8.568 Set.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Set.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:11
 */

#ifndef SET_H
#define SET_H
```

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/ParserChangesInformation.h"
#include "../../kernel/simulator/PluginInformation.h"
#include "../../kernel/simulator/EntityType.h"

class Set : public ModelElement {
public:
    Set(Model* model, std::string name = "");
    virtual ~Set() = default;
public: // static
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
    static PluginInformation* GetPluginInformation();
public:
    virtual std::string show();
public:
    void setSetOfType(std::string _setOfType);
    std::string getSetOfType() const;
    List<ModelElement*>* getElementSet() const;

protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
protected: // could be overriden by derived classes
    virtual bool _check(std::string* errorMessage);
    virtual ParserChangesInformation* _getParserChangesInformation();
private:
    //ElementManager* _elems;

    const struct DEFAULT_VALUES {
        unsigned int membersSize = 0;
        std::string setOfType = Util::TypeOf<EntityType>();
    } DEFAULT;
    List<ModelElement*>* _elementSet = new List<ModelElement*>();
    std::string _setOfType = DEFAULT.setOfType;
};

#endif /* SET_H */

```

8.569 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Station.cpp File Reference

```
#include "Station.h"
#include "../../kernel/simulator/Entity.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"
```

Include dependency graph for Station.cpp:

8.570 Station.cpp

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Station.cpp
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:12
 */

#include "Station.h"
#include "../../kernel/simulator/Entity.h"
#include "../../kernel/simulator/Model.h"
#include "../../kernel/simulator/Attribute.h"

Station::Station(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Station>(), name) {
}

Station::~Station() {
    //_parentModel->elements()->remove(Util::TypeOf<StatisticsCollector>(), _cstatNumberInStation);
    //_parentModel->elements()->remove(Util::TypeOf<StatisticsCollector>(), _cstatTimeInStation);
}

std::string Station::show() {
    std::string msg = ModelElement::show() + ",enterIntoStationComponent=";
    if (_enterIntoStationComponent == nullptr)
        msg += "NULL";
    else
        msg += _enterIntoStationComponent->getName();
    return msg;
}

void Station::initBetweenReplications() {
    _cstatNumberInStation->getStatistics()->getCollector()->clear();
    _cstatTimeInStation->getStatistics()->getCollector()->clear();

}

void Station::enter(Entity* entity) {
    std::string attributeName = "Entity.ArrivalAt" + this->getName();
    trimwithin(attributeName);
    entity->setAttributeValue(attributeName, _parentModel->getSimulation()->getSimulatedTime());
    entity->setAttributeValue("Entity.Station", _id);
    _numberInStation++;
    if (_reportStatistics)
        this->_cstatNumberInStation->getStatistics()->getCollector()->addValue(_numberInStation);
}

void Station::leave(Entity* entity) {
    std::string attributeName = "Entity.ArrivalAt" + this->getName();
    trimwithin(attributeName);
    double arrivalTime = entity->getAttributeValue(attributeName);
    double timeInStation = _parentModel->getSimulation()->getSimulatedTime() - arrivalTime;
    entity->setAttributeValue("Entity.Station", 0.0);
    _numberInStation--;
    if (_reportStatistics) {
        _cstatNumberInStation->getStatistics()->getCollector()->addValue(_numberInStation);
        _cstatTimeInStation->getStatistics()->getCollector()->addValue(timeInStation);
        if (entity->getEntityType()->isReportStatistics())
            entity->getEntityType()->addGetStatisticsCollector(entity->getEntityTypeName() + ".TimeInStations")->getStatistics()->getCollector()->addValue(timeInStation); // \todo: should check if entitytype reports (?)
    }
}

void Station::setEnterIntoStationComponent(ModelComponent* _enterIntoStationComponent) {
    this->_enterIntoStationComponent = _enterIntoStationComponent;
}

ModelComponent* Station::getEnterIntoStationComponent() const {
    return _enterIntoStationComponent;
}

PluginInformation* Station::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Station>(), &Station::LoadInstance);
    return info;
}

ModelElement* Station::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Station* newElement = new Station(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

```

```
    }
    return newElement;
}

bool Station::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* Station::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<Station>());
    return fields;
}

bool Station::_check(std::string* errorMessage) {
    /* include attributes needed */
    std::vector<std::string> neededNames = {"Entity.Station"};
    neededNames.insert(neededNames.begin(), "Entity.ArrivalAt" + this->getName());
    std::string neededName;
    for (unsigned int i = 0; i < neededNames.size(); i++) {
        neededName = neededNames[i];
        if (_parentModel->getElements()->getElement(Util::TypeOf<Attribute>(), neededName) == nullptr)
 {
            new Attribute(_parentModel, neededName);
        }
    }
    *errorMessage += "";
    return true;
}

void Station::_createInternalElements() {
    if (_reportStatistics) {
        if (_cstatNumberInStation == nullptr) {
            _cstatNumberInStation = new StatisticsCollector(_parentModel, getName() + "." +
 "NumberInStation", this);
            _cstatTimeInStation = new StatisticsCollector(_parentModel, getName() + "." +
 "TimeInStation", this);
            _childrenElements->insert({"NumberInStation", _cstatNumberInStation});
            _childrenElements->insert({"TimeInStation", _cstatTimeInStation});
            //
            // include StatisticsCollector needed in EntityType
            std::list<ModelElement*>* enttypes =
 _parentModel->getElements()->getElementList(Util::TypeOf<EntityType>())->list();
            for (ModelElement* element : *enttypes) {
                if (element->isReportStatistics())
                    static_cast<EntityType*> (element)->addGetStatisticsCollector(element->getName() +
 ".TimeInStations"); // force create this CStat before simulation starts
            }

        }
    } else
        if (_cstatNumberInStation != nullptr) {
        _removeChildrenElements();
    }
}
```

8.571 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Station.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Plugin.h"
#include "../../kernel/simulator/Entity.h"
```

Include dependency graph for Station.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Station

8.572 Station.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Station.h
 * Author: rlcancian
 *
 * Created on 03 de Junho de 2019, 15:12
 */

#ifndef STATION_H
#define STATION_H

#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Plugin.h"
#include "../../kernel/simulator/Entity.h"

/*
 Station module
DESCRIPTION
The Station module defines a station (or a set of stations) corresponding to a physical
or logical location where processing occurs. If the Station module defines a station
set, it is effectively defining multiple processing locations.
The station (or each station within the defined set) has a matching Activity Area that
is used to report all times and costs accrued by the entities in this station. This
Activity Area's name is the same as the station. If a parent Activity Area is defined,
then it also accrues any times and costs by the entities in this station.
TYPICAL USES
 Defining a lathe area
 Defining a set of toll booths
 Defining a food preparation area
PROMPTS
Prompt Description
Name Unique name of the module that will be displayed in the
flowchart.
```

```
Station Type Type of station being defined, either as an individual Station or a
station Set.
Station Name Name of the individual station.
Set Name Name of the station set.
Parent Activity Area Name of the Activity Area's parent.
Associated Intersection Name of the intersection associated with this station in a guided
transporter network.
Report Statistics Specifies whether or not statistics will automatically be collected
and stored in the report database for this station and its
corresponding activity area.
Save Attribute Attribute name used to store the index number into the station set
of the member that is selected.
Station Set Members Names of the stations that are members of this station set.
Station Name A given station can only exist once within a model. Therefore, an
individual station can only be the member of one station set, and
that individual station may not be the name of a station in
another module.
Parent Activity Area Name of the Activity Area's parent for the station set member.
Associated Intersection Name of the intersection associated with this station set in a
guided transporter network.
Report Statistics Specifies whether or not statistics will automatically be collected
and stored in the report database for this station set member and
its corresponding activity area.
 */
class Station : public ModelElement {
public:
    Station(Model* model, std::string name = "");
    virtual ~Station();
public:
    virtual std::string show();
public: // static
    static PluginInformation* GetPluginInformation();
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
public:
    void initBetweenReplications();
    void enter(Entity* entity);
    void leave(Entity* entity);
    void setEnterIntoStationComponent(ModelComponent* _enterIntoStationComponent);
    ModelComponent* getEnterIntoStationComponent() const;
protected:
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
    virtual bool _check(std::string* errorMessage);
    virtual void _createInternalElements();
private:
    unsigned int _numberInStation = 0;
    ModelComponent* _enterIntoStationComponent;
private: // inner elements
    StatisticsCollector* _cstatNumberInStation = nullptr;
    StatisticsCollector* _cstatTimeInStation;
};

#endif /* STATION_H */

```

8.573 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Storage.cpp File Reference

`#include "Storage.h"`

Include dependency graph for Storage.cpp:

8.574 Storage.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Storage.cpp
 * Author: rlcancian
 *
 * Created on 20 de Storageembro de 2019, 20:06
 */

#include "Storage.h"

Storage::Storage(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Storage>(), name) {
}

std::string Storage::show() {
    return ModelElement::show() +
            "";
}

PluginInformation* Storage::GetPluginInformation() {
    PluginInformation* info = new PluginInformation(Util::TypeOf<Storage>(), &Storage::LoadInstance);
    return info;
}

ModelElement* Storage::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
    Storage* newElement = new Storage(model);
    try {
        newElement->_loadInstance(fields);
    } catch (const std::exception& e) {

    }
    return newElement;
}

bool Storage::_loadInstance(std::map<std::string, std::string>* fields) {
    bool res = ModelElement::_loadInstance(fields);
    if (res) {
        try {
            //this->_attributeName = (*fields->find("attributeName")).second;
            //this->_orderRule = static_cast<OrderRule> (std::stoi((*fields->find("orderRule")).second));
        } catch (...) {
        }
    }
    return res;
}

std::map<std::string, std::string>* Storage::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance(); //Util::TypeOf<Storage>());
    //SaveField(fields, "orderRule", std::to_string(static_cast<int> (this->_orderRule)));
    //SaveField(fields, "attributeName", "\""+this->_attributeName+"\"");
    return fields;
}

bool Storage::_check(std::string* errorMessage) {
    bool resultAll = true;
    // resultAll |= ...
    *errorMessage += "";
    return resultAll;
}

ParserChangesInformation* Storage::_getParserChangesInformation() {
    ParserChangesInformation* changes = new ParserChangesInformation();
    //changes->getProductionToAdd()->insert(...);
    //changes->getTokensToAdd()->insert(...);
    return changes;
}

```

8.575 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Storage.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
```

```
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/PluginInformation.h"
```

Include dependency graph for Storage.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Storage

8.576 Storage.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Storage.h
 * Author: rlcancian
 *
 * Created on 20 de Storageembro de 2019, 20:06
 */

#ifndef STORAGE_H
#define STORAGE_H


#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
//#include "../../kernel/simulator/ParserChangesInformation.h"
#include "../../kernel/simulator/PluginInformation.h"

/*
Storage module
DESCRIPTION
The Storage module defines the name of a storage. Storages are automatically created
by any module that references the storage so that this module is seldom needed. The
only time this module is needed is when a storage is defined as a member of a storage
set or specified using an attribute or expression.
TYPICAL USES
```

```
 Defining an animate storage for a set of storages
PROMPTS
Prompt Description
Name The name of the storage set being defined. This name must be
unique.
 */
class Storage : public ModelElement {
public:
    Storage(Model* model, std::string name = "");
    virtual ~Storage() = default;
public: // static
    static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
    static PluginInformation* GetPluginInformation();
public:
    virtual std::string show();

protected: // must be overriden by derived classes
    virtual bool _loadInstance(std::map<std::string, std::string>* fields);
    virtual std::map<std::string, std::string>* _saveInstance();
protected: // could be overriden by derived classes
    virtual bool _check(std::string* errorMessage);
    virtual ParserChangesInformation* _getParserChangesInformation();
};
#endif /* STORAGE_H */

```

8.577 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Variable.cpp File Reference

```
#include "Variable.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Variable.cpp:

8.578 Variable.cpp

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Variable.cpp
 * Author: rafael.luiz.cancian
 *
 * Created on 4 de Setembro de 2018, 18:28
 */

#include "Variable.h"
#include "../../kernel/simulator/Plugin.h"

Variable::Variable(Model* model, std::string name) : ModelElement(model, Util::TypeOf<Variable>(),
    name) {
    setName(name);
}
```

```
00020
00021 std::string Variable::show() {
00022     std::string text = "values:{";
00023     for (std::pair<std::string, double> var : * this->_values) {
00024         text += var.first + "=" + strTruncIfInt(std::to_string(var.second)) + ", ";
00025     }
00026     text = text.substr(0, text.length() - 2);
00027     text += "}";
00028     return ModelElement::show();
00029 }
00030
00031 PluginInformation* Variable::GetPluginInformation() {
00032     PluginInformation* info = new PluginInformation(Util::TypeOf<Variable>(),
       &Variable::LoadInstance);
00033     return info;
00034 }
00035
00036 double Variable::value() {
00037     return value("");
00038 }
00039
00040 double Variable::value(std::string index) {
00041     std::map<std::string, double>::iterator it = _values->find(index);
00042     if (it == _values->end()) {
00043         return 0.0; // index does not exist. Assuming sparse matrix, it's zero.
00044     } else {
00045         return it->second;
00046     }
00047 }
00048
00049 void Variable::setValue(double value) {
00050     setValue("", value);
00051 }
00052
00053 void Variable::setValue(std::string index, double value) {
00054     std::map<std::string, double>::iterator it = _values->find(index);
00055     if (it == _values->end()) {
00056         // index does not exist. Create it.
00057         _values->insert({index, value}); //(std::pair<std::string, double>(index, value));
00058     } else {
00059         it->second = value;
00060     }
00061 }
00062
00063 double Variable::initialValue() {
00064     return initialValue("");
00065 }
00066
00067 void Variable::setInitialValue(double value) {
00068     setInitialValue("", value);
00069 }
00070
00071 double Variable::initialValue(std::string index) {
00072     std::map<std::string, double>::iterator it = _initialValues->find(index);
00073     if (it == _initialValues->end()) {
00074         return 0.0; // index does not exist. Assuming sparse matrix, it's zero.
00075     } else {
00076         return it->second;
00077     }
00078 }
00079
00080 void Variable::setInitialValue(std::string index, double value) {
00081     std::map<std::string, double>::iterator it = _initialValues->find(index);
00082     if (it == _initialValues->end()) {
00083         // index does not exist. Create it.
00084         _initialValues->insert(std::pair<std::string, double>(index, value));
00085     } else {
00086         it->second = value;
00087     }
00088 }
00089
00090 List<unsigned int>* Variable::dimensionSizes() const {
00091     return _dimensionSizes;
00092 }
00093
00094 ModelElement* Variable::LoadInstance(Model* model, std::map<std::string, std::string>* fields) {
00095     Variable* newElement = new Variable(model);
00096     try {
00097         newElement->_loadInstance(fields);
00098     } catch (const std::exception& e) {
00099
00100     }
00101     return newElement;
00102 }
00103
00104 bool Variable::_loadInstance(std::map<std::string, std::string>* fields) {
00105     bool res = ModelElement::_loadInstance(fields);
```

```
    if (res) {
        unsigned int nv = LoadField(fields, "numValues", 0);
        std::string pos;
        double value;
        for (unsigned int i = 0; i < nv; i++) {
            pos = LoadField(fields, "pos" + std::to_string(i), 0);
            value = LoadField(fields, "value" + std::to_string(i), 0);
            this->_initialValues->emplace(pos, value);
        }
    }
    return res;
}

std::map<std::string, std::string>* Variable::_saveInstance() {
    std::map<std::string, std::string>* fields = ModelElement::_saveInstance();
 //Util::TypeOf<Variable>());
    SaveField(fields, "numValues", _initialValues->size(), 0);
    unsigned int i = 0;
    for (std::map<std::string, double>::iterator it = this->_initialValues->begin(); it !=
 _initialValues->end(); it++, i++) {
        SaveField(fields, "pos" + std::to_string(i), (*it).first, 0);
        SaveField(fields, "value" + std::to_string(i), (*it).second, 0.0);
    }
    return fields;
}

bool Variable::_check(std::string* errorMessage) {
    *errorMessage += "";
    return true;
}

void Variable::_initBetweenReplications() {
    this->_values->clear();
    this->_values = this->_initialValues;
}
```

8.579 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/plugins/elements/Variable.h File Reference

```
#include "../../kernel/simulator/ModelElement.h"
#include "../../kernel/simulator/ElementManager.h"
#include "../../kernel/simulator/Plugin.h"
```

Include dependency graph for Variable.h:

This graph shows which files directly or indirectly include this file:

Classes

- class Variable

8.580 Variable.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Variable.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 4 de Setembro de 2018, 18:28
00012  */
00013
00014 #ifndef VARIABLE_H
00015 #define VARIABLE_H
00016
00017 #include "../../kernel/simulator/ModelElement.h"
00018 #include "../../kernel/simulator/ElementManager.h"
00019 #include "../../kernel/simulator/Plugin.h"
00020
00090 class Variable : public ModelElement {
00091 public:
00092     Variable(Model* model, std::string name = "");
00093     virtual ~Variable() = default;
00094 public:
00095     virtual std::string show();
00096 public: //static
00097     static PluginInformation* GetPluginInformation();
00098     static ModelElement* LoadInstance(Model* model, std::map<std::string, std::string>* fields);
00099 public:
00100     double value();
00101     void setValue(double value);
00102     double value(std::string index);
00103     void setValue(std::string index, double value);
00104     double initialValue();
00105     void setInitialValue(double value);
00106     double initialValue(std::string index);
00107     void setInitialValue(std::string index, double value);
00108     List<unsigned int>* dimensionSizes() const;
00109
00110 protected:
00111     virtual bool _loadInstance(std::map<std::string, std::string>* fields);
00112     virtual std::map<std::string, std::string>* _saveInstance();
00113     virtual bool _check(std::string* errorMessage);
00114     virtual void _initBetweenReplications();
00115
00116 private:
00117     List<unsigned int>* _dimensionSizes = new List<unsigned int>();
00118     std::map<std::string, double>* _values = new std::map<std::string, double>();
00119     std::map<std::string, double>* _initialValues = new std::map<std::string, double>();
00120 };
00121
00122 #endif /* VARIABLE_H */
00123
```

8.581 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tests/testProbabilityDistribution.cpp File Reference

```
#include <stdlib.h>
#include <iostream>
#include "../tools/ProbabilityDistribution.h"
#include "../tools/solver_if.h"
#include "../tools/SolverDefaultImpl1.h"
#include "../kernel/statistics/SamplerDefaultImpl1.h"
#include "../kernel/statistics/CollectorDatafileDefaultImpl1.h"
#include "../kernel/statistics/StatisticsDataFileDefaultImpl.h"
#include "../Traits.h"
#include "../tools/HypothesisTester_if.h"
```

Include dependency graph for testProbabilityDistribution.cpp:

Functions

- void test1 ()
- void test2 ()
- void test3 ()
- void test4 ()
- void test5 ()
- int mainn (int argc, char **argv)

8.581.1 Function Documentation

8.581.1.1 mainn()

```
int mainn (
    int argc,
    char ** argv )
```

Definition at line 94 of file testProbabilityDistribution.cpp.

Here is the call graph for this function:

8.581.1.2 test1() `void test1 ( )`

Definition at line 29 of file testProbabilityDistribution.cpp.

Here is the call graph for this function:

8.581.1.3 test2() `void test2 ( )`

Definition at line 56 of file testProbabilityDistribution.cpp.

Here is the call graph for this function:

8.581.1.4 test3() `void test3 ( )`

Definition at line 64 of file testProbabilityDistribution.cpp.

Here is the call graph for this function:

8.581.1.5 test4() `void test4 ( )`

Definition at line 76 of file testProbabilityDistribution.cpp.

Here is the call graph for this function:

8.581.1.6 test5() `void test5 ( )`

Definition at line 88 of file testProbabilityDistribution.cpp.

Here is the call graph for this function:

Here is the caller graph for this function:

8.582 testProbabilityDistribution.cpp

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/cppsimpletest.cc to edit this template
 */

/*
 * File:   testProbabilityDistribution.cpp
 * Author: rlcancian
 *
 * Created on 23 de novembro de 2021, 18:30
 */

#include <stdlib.h>
#include <iostream>

/*
 * Simple C++ Test Suite
 */

#include "../tools/ProbabilityDistribution.h"
#include "../tools/solver_if.h"
#include "../tools/SolverDefaultImpl1.h"
#include "../kernel/statistics/SamplerDefaultImpl1.h"
#include "../kernel/statistics/CollectorDatafileDefaultImpl1.h"
#include "../kernel/statistics/StatisticsDataFileDefaultImpl.h"
#include "../Traits.h"
#include "../tools/HypothesisTester_if.h"

void test1() {
    //std::cout « "testProbabilityDistribution test 1" « std::endl;

    Sampler_if* sampler = new SamplerDefaultImpl1();
    StatisticsDatafile_if *stat = new StatisticsDatafileDefaultImpl1();
    CollectorDatafile_if *collector = static_cast<CollectorDatafile_if*> (stat->getCollector());
    collector->setDataFilename("./datafile.txt");
    double x;
    for (unsigned short i = 0; i < 5e3; i++) {
        x = sampler->sampleNormal(100, 20);
        collector->addValue(x);
    }
    std::cout « "min: " « stat->min() « std::endl;
    std::cout « "max: " « stat->max() « std::endl;
    std::cout « "avg: " « stat->average() « std::endl;
    std::cout « "std: " « stat->stddeviation() « std::endl;
    std::cout « "e0: " « stat->halfWidthConfidenceInterval() « std::endl;
    std::cout « "med: " « stat->mediane() « std::endl;
    std::cout « "mod: " « stat->mode() « std::endl;
    for (unsigned short i = 1; i <= 4; i++) {
        std::cout « "qrt" « i « ": " « stat->quartil(i) « std::endl;
    }
    for (unsigned short i = 1; i <= 10; i++) {
        std::cout « "dec" « i « ": " « stat->decil(i) « std::endl;
    }
}

void test2() {
    //std::cout « "testProbabilityDistribution test 2" « std::endl;
    //std::cout « "%TEST_FAILED% time=0 testname=test2 (testProbabilityDistribution) message=error
 message sample" « std::endl;
    Solver_if* integr = new SolverDefaultImpl1(1e-6, 1e3);
    double x = integr->integrate(0.0, 0.5, &ProbabilityDistribution::beta, 4.0, 2.0);
    std::cout « x « std::endl;
}

void test3() {
    double x = ProbabilityDistribution::inverseNormal(0.975, 0.0, 1.0);
    std::cout « x « std::endl;
    x = ProbabilityDistribution::inverseTStudent(0.975, 0.0, 1.0, 18);
    x = ProbabilityDistribution::inverseTStudent(0.975, 0.0, 1.0, 18);
    std::cout « x « std::endl;
    x = ProbabilityDistribution::inverseFFisherSnedecor(0.975, 2.0, 1.0);
    std::cout « x « std::endl;
    x = ProbabilityDistribution::inverseChi2(0.975, 5);
    std::cout « x « std::endl;
}

void test4() {
    HypothesisTester_if* ht = new Traits<HypothesisTester_if>::Implementation();
    HypothesisTester_if::ConfidenceInterval ic = ht->averageConfidenceInterval(100, 10, 200, 0.95);
    double x;
    x = ProbabilityDistribution::inverseFFisherSnedecor(0.05, 7, 16);
    x = ProbabilityDistribution::inverseFFisherSnedecor(0.025, 7, 16);
    x = ProbabilityDistribution::inverseFFisherSnedecor(1.0 - 0.05, 16, 7);
    x = ProbabilityDistribution::inverseFFisherSnedecor(1.0 - 0.025, 16, 7);
    ic = ht->varianceRatioConfidenceInterval(46.5, 8, 23.4, 17, 0.95);
```

```
    std::cout « "[" « ic.inferiorLimit() « "," « ic.superiorLimit() « "]" « std::endl;
}

void test5() {
    HypothesisTester_if* ht = new Traits<HypothesisTester_if>::Implementation();
    HypothesisTester_if::TestResult result = ht->testAverage(90.0, 10.0, 25, 84.0, 0.95,
    HypothesisTester_if::H1Comparition::LESS_THAN);
    std::cout « result.errorTypeI() « result.pValue() « result.rejectH0() « std::endl;
}

int mainn(int argc, char** argv) {
    std::cout « "%SUITE_STARTING% testProbabilityDistribution" « std::endl;
    std::cout « "%SUITE_STARTED%" « std::endl;

    //test1();
    //test2();
    //test3();
    //test4();
    test5();
    // std::cout « "%SUITE_FINISHED% time=0" « std::endl;

    return (EXIT_SUCCESS);
}
```

8.583 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/HypothesisTester_if.h File Reference

`#include <string>`

Include dependency graph for HypothesisTester_if.h:

This graph shows which files directly or indirectly include this file:

Classes

- class HypothesisTester_if
- class HypothesisTester_if::ConfidenceInterval
- class HypothesisTester_if::TestResult

Typedefs

- typedef bool(∗ checkProportionFunction) (double value)

8.583.1 Typedef Documentation

8.583.1.1 checkProportionFunction `typedef bool(* checkProportionFunction) (double value)`

Definition at line 19 of file HypothesisTester_if.h.

8.584 HypothesisTester_if.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   HypothesisTester_if.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 23 de Agosto de 2018, 19:04
00012  */
00013
00014 #ifndef HYPOTHESISTESTER_IF_H
00015 #define HYPOTHESISTESTER_IF_H
00016
00017 #include <string>
00018
00019 typedef bool (*checkProportionFunction)(double value);
00020
00025 class HypothesisTester_if {
00026 public:
00027
00028     class ConfidenceInterval {
00029     public:
00030         ConfidenceInterval(double inferiorLimit, double superiorLimit) {
00031             _infLim = inferiorLimit;
00032             _supLim = superiorLimit;
00033         }
00034         double inferiorLimit() {
00035             return _infLim;
00036         }
00037         double superiorLimit() {
00038             return _supLim;
00039         }
00040
00041     private:
00042         double _infLim, _supLim;
00043     };
00044
00045     enum H1Comparition {
00046         DIFFERENT = 1,
00047         LESS_THAN = 2,
00048         GREATER_THAN = 3
00049     };
00050
00051     class TestResult {
00052     public:
00053         TestResult(double errorTypeI, double pvalue, bool rejectH0, double rejectionThreshold, double
      testStat) {
```

```
            _errorTypeI = errorTypeI;
            _pvalue = pvalue;
            _rejectH0 = rejectH0;
            _rejectionThreshold = rejectionThreshold;
            _testStat = testStat;
        }
        bool rejectH0() {
            return _rejectH0;
        }
        double pValue() {
            return _pvalue;
        }
        double errorTypeI() {
            return _errorTypeI;
        }
        double rejectionThreshold() {
            return _rejectionThreshold;
        }
        double testStat() {
            return _testStat;
        }

    private:
        double _pvalue, _confidenceLevel, _errorTypeI, _rejectionThreshold, _testStat;
        bool _rejectH0;
    };
public:
    // confidence intervals
    virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval(double avg, double stddev, unsigned int n, double confidenceLevel) = 0;
    virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval(double prop, unsigned int n, double confidenceLevel) = 0;
    virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval(double var, unsigned int n, double confidenceLevel) = 0;
    virtual HypothesisTester_if::ConfidenceInterval averageDifferenceConfidenceInterval(double avg1, double stddev1, unsigned int n1, double avg2, double stddev2, unsigned int n2, double confidenceLevel) = 0;
    virtual HypothesisTester_if::ConfidenceInterval varianceRatioConfidenceInterval(double var1, unsigned int n1, double var2, unsigned int n2, double confidenceLevel) = 0;
    // confidence intervals based on datafile
    virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval(std::string sampleDataFilename, double confidenceLevel) = 0;
    virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval(std::string sampleDataFilename, checkProportionFunction function, double confidenceLevel) = 0;
    virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval(std::string sampleDataFilename, double confidenceLevel) = 0;
    // parametric tests
    // one population
    virtual HypothesisTester_if::TestResult testAverage(double avg, double stddev, unsigned int n, double avgSample, double confidenceLevel, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testProportion(double prop, unsigned int n, double proptest, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testVariance(double var, unsigned int n, double vartest, HypothesisTester_if::H1Comparition comp) = 0;
    // two populations
    virtual HypothesisTester_if::TestResult testAverage(double avg1, unsigned int n1, double avg2, unsigned int n2, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testProportion(double prop1, unsigned int n1, double prop2, unsigned int n2, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testVariance(double var1, unsigned int n1, double var2, unsigned int n2, HypothesisTester_if::H1Comparition comp) = 0;
    // one population based on datafile
    virtual HypothesisTester_if::TestResult testAverage(std::string sampleDataFilename, double avgSample, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testProportion(std::string sampleDataFilename, checkProportionFunction function, double proptest, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testVariance(std::string sampleDataFilename, double vartest, HypothesisTester_if::H1Comparition comp) = 0;
    // two populations based on datafile
    virtual HypothesisTester_if::TestResult testAverage(std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testProportion(std::string firstSampleDataFilename, std::string secondSampleDataFilename, checkProportionFunction function, HypothesisTester_if::H1Comparition comp) = 0;
    virtual HypothesisTester_if::TestResult testVariance(std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp) = 0;
    // @TODO: Add interface for non-parametrical tests, such as chi-square (based on values and on datafile)
};

#endif /* HYPOTHESISTESTER_IF_H */
```

8.585 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tools/HypothesisTesterDefault↩ Impl1.cpp File Reference

```
#include "HypothesisTesterDefaultImpl1.h"
#include "ProbabilityDistribution.h"
#include "../kernel/statistics/StatisticsDataFile_if.h"
#include "../kernel/TraitsKernel.h"
#include <math.h>
```

Include dependency graph for HypothesisTesterDefaultImpl1.cpp:

8.586 HypothesisTesterDefaultImpl1.cpp

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.cc to edit this template
 */

/*
 * File:   HypothesisTesterDefaultImpl1.cpp
 * Author: rlcancian
 *
 * Created on 24 de novembro de 2021, 02:52
 */

#include "HypothesisTesterDefaultImpl1.h"
#include "ProbabilityDistribution.h"
#include "../kernel/statistics/StatisticsDataFile_if.h"
#include "../kernel/TraitsKernel.h"
#include <math.h>

HypothesisTesterDefaultImpl1::HypothesisTesterDefaultImpl1() {
}

// confidence intervals

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::averageConfidenceInterval(double avg, double stddev, unsigned int n, double confidenceLevel) {
    double correctConf = (1.0 - confidenceLevel) / 2.0;
    double e0 = -ProbabilityDistribution::inverseNormal(correctConf, 0.0, 1.0) * (stddev / sqrt(n));
    return HypothesisTester_if::ConfidenceInterval(avg - e0, avg + e0);
}

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::proportionConfidenceInterval(double prop, unsigned int n, double confidenceLevel) {
    double correctConf = (1.0 - confidenceLevel) / 2.0;
    double e0 = -ProbabilityDistribution::inverseNormal(correctConf, 0.0, 1.0) * sqrt(prop * (1 - prop) / n);
    return HypothesisTester_if::ConfidenceInterval(prop - e0, prop + e0);
}

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::varianceConfidenceInterval(double var, unsigned int n, double confidenceLevel) {
    double il = (n - 1) * var / ProbabilityDistribution::inverseChi2((1.0 - confidenceLevel) / 2.0, n - 1);
    double sl = (n - 1) * var / ProbabilityDistribution::inverseChi2(confidenceLevel / 2.0, n - 1);
    return HypothesisTester_if::ConfidenceInterval(il, sl);
}

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::averageDifferenceConfidenceInterval(double avg1, double stddev1, unsigned int n1, double avg2, double stddev2, unsigned int n2, double confidenceLevel) {
    double correctConf = (1.0 - confidenceLevel) / 2.0;
    double e0;
    HypothesisTester_if::ConfidenceInterval varIC = varianceRatioConfidenceInterval(pow(stddev1, 2), n1, pow(stddev2, 2), n2, confidenceLevel);
```

```
    if ((varIC.inferiorLimit() <= 1.0 && varIC.superiorLimit() >= 1.0) || (varIC.inferiorLimit() >= 1.0 && varIC.superiorLimit() <= 1.0)) { // test variances ratio
        // equal variances
        e0 = -ProbabilityDistribution::tStudent(correctConf, 0.0, 1.0, n1 + n2 - 2) * sqrt(pow(stddev1, 2) * pow(stddev2, 2) * (1 / n1 + 1 / n2));
    } else { // different variances
        double degreeFreedom = pow(pow(stddev1, 2) / n1 + pow(stddev2, 2) / n2, 2) / (pow(pow(stddev1, 2) / n1, 2) / (n1 + 1) + pow(pow(stddev2, 2) / n2, 2) / (n2 + 1)) - 2;
        e0 = -ProbabilityDistribution::tStudent(correctConf, 0.0, 1.0, degreeFreedom) * sqrt(pow(stddev1, 2) / n1 + pow(stddev2, 2) / n2);
    }
    return HypothesisTester_if::ConfidenceInterval(avg1 - avg2 - e0, avg1 - avg2 + e0);
}

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::varianceRatioConfidenceInterval(double var1, unsigned int n1, double var2, unsigned int n2, double confidenceLevel) {
    double ratio = var1 / var2;
    double il = 1 / ProbabilityDistribution::inverseFFisherSnedecor((1.0 - confidenceLevel) / 2.0, n2 - 1, n1 - 1);
    il *= ratio;
    double sl = ProbabilityDistribution::inverseFFisherSnedecor(confidenceLevel / 2.0, n1 - 1, n2 - 1);
    sl *= ratio;
    return HypothesisTester_if::ConfidenceInterval(il, sl);
}


// confidence intervals based on datafile

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::averageConfidenceInterval(std::string sampleDataFilename, double confidenceLevel) {
    StatisticsDatafile_if* stat = new TraitsKernel<StatisticsDatafile_if>::Implementation();
    static_cast<CollectorDatafile_if*> (stat->getCollector())->setDataFilename(sampleDataFilename);
    return averageConfidenceInterval(stat->average(), stat->stddeviation(), stat->numElements(), confidenceLevel);
}

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::proportionConfidenceInterval(std::string sampleDataFilename, checkProportionFunction function, double confidenceLevel) {
    StatisticsDatafile_if* stat = new TraitsKernel<StatisticsDatafile_if>::Implementation();
    static_cast<CollectorDatafile_if*> (stat->getCollector())->setDataFilename(sampleDataFilename);
    unsigned long count = 0;
    double value;
    for (unsigned long i = 0; i < stat->numElements(); i++) {
        value = static_cast<CollectorDatafile_if*> (stat->getCollector())->getValue(i);
        if (function(value))
            count++;
    }
    double prop = (double) count / stat->numElements();
    return proportionConfidenceInterval(prop, stat->numElements(), confidenceLevel);
}

HypothesisTester_if::ConfidenceInterval HypothesisTesterDefaultImpl1::varianceConfidenceInterval(std::string sampleDataFilename, double confidenceLevel) {
    StatisticsDatafile_if* stat = new TraitsKernel<StatisticsDatafile_if>::Implementation();
    static_cast<CollectorDatafile_if*> (stat->getCollector())->setDataFilename(sampleDataFilename);
    return varianceConfidenceInterval(stat->variance(), stat->numElements(), confidenceLevel);
}
// parametric tests
// one population

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testAverage(double avg, double stddev, unsigned int n, double avgSample, double confidenceLevel, HypothesisTester_if::H1Comparition comp) {
    double significanceLevel = (1.0 - confidenceLevel);
    double cumulative;
    if (comp == HypothesisTester_if::H1Comparition::LESS_THAN) {
        cumulative = significanceLevel;
    } else if (comp == HypothesisTester_if::H1Comparition::GREATER_THAN) {
        cumulative = confidenceLevel;
    } else {
        cumulative = confidenceLevel + significanceLevel / 2.0;
    }
    double threshold = ProbabilityDistribution::inverseTStudent(cumulative, 0.0, 1.0, n - 1);
    double testStat = (avgSample - avg) / (stddev / sqrt(n));
    double pvalue = ProbabilityDistribution::tStudent(avgSample, avg, stddev, n - 1);
    return HypothesisTester_if::TestResult(significanceLevel, pvalue, pvalue<significanceLevel, threshold, testStat);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testProportion(double prop, unsigned int n, double proptest, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}
```

```
HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testVariance(double var, unsigned int n, double vartest, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}
// two populations

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testAverage(double avg1, unsigned int n1, double avg2, unsigned int n2, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testProportion(double prop1, unsigned int n1, double prop2, unsigned int n2, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testVariance(double var1, unsigned int n1, double var2, unsigned int n2, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}
// one population based on datafile

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testAverage(std::string sampleDataFilename, double avgSample, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testProportion(std::string sampleDataFilename, checkProportionFunction function, double proptest, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testVariance(std::string sampleDataFilename, double vartest, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}
// two populations based on datafile

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testAverage(std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testProportion(std::string firstSampleDataFilename, std::string secondSampleDataFilename, checkProportionFunction function, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}

HypothesisTester_if::TestResult HypothesisTesterDefaultImpl1::testVariance(std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp) {
    return HypothesisTester_if::TestResult(0.0, 0.0, false, 0.0, 0.0);
}
// @TODO: Add interface for non-parametrical tests, such as chi-square (based on values and on datafile)
```

8.587 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tools/HypothesisTesterDefaultImpl1.h File Reference

```
#include "HypothesisTester_if.h"
```

Include dependency graph for HypothesisTesterDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class HypothesisTesterDefaultImpl1

8.588 HypothesisTesterDefaultImpl1.h

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.h to edit this template
 */

/*
 * File:   HypothesisTesterDefaultImpl1.h
 * Author: rlcancian
 *
 * Created on 24 de novembro de 2021, 02:52
 */

#ifndef HYPOTHESISTESTERDEFAULTIMPL1_H
#define HYPOTHESISTESTERDEFAULTIMPL1_H

#include "HypothesisTester_if.h"

class HypothesisTesterDefaultImpl1 : public HypothesisTester_if {
public:
    HypothesisTesterDefaultImpl1();
    virtual ~HypothesisTesterDefaultImpl1() = default;
```

```
public:
    // confidence intervals
    virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval(double avg, double stddev, unsigned int n, double confidenceLevel);
    virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval(double prop, unsigned int n, double confidenceLevel);
    virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval(double var, unsigned int n, double confidenceLevel);
    virtual HypothesisTester_if::ConfidenceInterval averageDifferenceConfidenceInterval(double avg1, double stddev1, unsigned int n1, double avg2, double stddev2, unsigned int n2, double confidenceLevel);
    virtual HypothesisTester_if::ConfidenceInterval varianceRatioConfidenceInterval(double var1, unsigned int n1, double var2, unsigned int n2, double confidenceLevel);

    // confidence intervals based on datafile
    virtual HypothesisTester_if::ConfidenceInterval averageConfidenceInterval(std::string sampleDataFilename, double confidenceLevel);
    virtual HypothesisTester_if::ConfidenceInterval proportionConfidenceInterval(std::string sampleDataFilename, checkProportionFunction function, double confidenceLevel);
    virtual HypothesisTester_if::ConfidenceInterval varianceConfidenceInterval(std::string sampleDataFilename, double confidenceLevel);
    // parametric tests
    // one population
    virtual HypothesisTester_if::TestResult testAverage(double avg, double stddev, unsigned int n, double avgSample, double confidenceLevel, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testProportion(double prop, unsigned int n, double proptest, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testVariance(double var, unsigned int n, double vartest, HypothesisTester_if::H1Comparition comp);
    // two populations
    virtual HypothesisTester_if::TestResult testAverage(double avg1, unsigned int n1, double avg2, unsigned int n2, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testProportion(double prop1, unsigned int n1, double prop2, unsigned int n2, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testVariance(double var1, unsigned int n1, double var2, unsigned int n2, HypothesisTester_if::H1Comparition comp);
    // one population based on datafile
    virtual HypothesisTester_if::TestResult testAverage(std::string sampleDataFilename, double avgSample, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testProportion(std::string sampleDataFilename, checkProportionFunction function, double proptest, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testVariance(std::string sampleDataFilename, double vartest, HypothesisTester_if::H1Comparition comp);
    // two populations based on datafile
    virtual HypothesisTester_if::TestResult testAverage(std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testProportion(std::string firstSampleDataFilename, std::string secondSampleDataFilename, checkProportionFunction function, HypothesisTester_if::H1Comparition comp);
    virtual HypothesisTester_if::TestResult testVariance(std::string firstSampleDataFilename, std::string secondSampleDataFilename, HypothesisTester_if::H1Comparition comp);
    // @TODO: Add interface for non-parametrical tests, such as chi-square (based on values and on datafile)
private:

};

#endif /* HYPOTHESISTESTERDEFAULTIMPL1_H */

```

8.589 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tools/ProbabilityDistribution.cpp File Reference

```
#include "ProbabilityDistribution.h"
#include "../Traits.h"
#include <math.h>
#include <cassert>
```

Include dependency graph for ProbabilityDistribution.cpp:

8.590 ProbabilityDistribution.cpp

```cpp
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.cc to edit this template
 */

/*
 * File:   ProbabilityDistribution.cpp
 * Author: rlcancian
 *
 * Created on 22 de novembro de 2021, 17:24
 */

#include "ProbabilityDistribution.h"
#include "../Traits.h"
#include <math.h>
#include <cassert>

std::map<std::string, double>* ProbabilityDistribution::memory = new std::map<std::string, double>();
Solver_if* ProbabilityDistribution::integrator = new Traits<Solver_if>::Implementation(Traits<Solver_if>::Precision, Traits<Solver_if>::MaxSteps);

double ProbabilityDistribution::beta(double x, double alpha, double beta) {
    double x1 = _gammaFunction(alpha + beta) / (_gammaFunction(alpha) * _gammaFunction(beta));
    //double x1 = 1/_betaFunction(alpha,beta);
    double x2 = pow(x, alpha - 1) * pow(1 - x, beta - 1);
    return x1*x2;
}

double ProbabilityDistribution::chi2(double x, double degreeFreedom) {
    if (x >= 0) {
        double x1 = pow(x, degreeFreedom / 2 - 1) * pow(M_E, -x / 2);
        double x2 = pow(2, degreeFreedom / 2) * _gammaFunction(degreeFreedom / 2);
        return x1 / x2;
    } else
        return 0.0;
}

double ProbabilityDistribution::erlang(double x, double shape, double scale) {
    assert(x >= 0.0 && scale >= 0.0);
    double x1 = pow(x, shape - 1) * pow(M_E, -x / scale);
    double x2 = pow(scale, shape)*(shape - 1);
    return x1 / x2;
} // int M

double ProbabilityDistribution::exponential(double x, double mean) {
    if (x >= 0) {
        return mean * pow(M_E, -mean * x);
    } else
        return 0.0;
}

double ProbabilityDistribution::fisherSnedecor(double x, double d1, double d2) {
    assert(x >= 0);
    double x1 = 1 / _betaFunction(d1 / 2, d2 / 2);
    double x2 = pow(d1 / d2, d1 / 2) * pow(x, d1 / 2 - 1);
    double x3 = pow(1 + (d1 / d2) * x, -(d1 + d2) / 2);
    return x1 * x2 * x3;
}

double ProbabilityDistribution::gamma(double x, double shape, double scale) {
    double x1 = pow(x, shape - 1) * pow(M_E, -x / scale);
    double x2 = pow(scale, shape) * _gammaFunction(shape);
    return x1 / x2;
}

double ProbabilityDistribution::logNormal(double x, double mean, double stddev) {
    double x1 = 1 / (x * stddev * sqrt(2 * M_PI));
    double x2 = exp(-pow(log(x) - mean, 2) / (2 * stddev * stddev));
    return x1 * x2;
}

double ProbabilityDistribution::normal(double x, double mean, double stddev) {
    double x1 = 1 / (stddev * sqrt(2 * M_PI));
    double x2 = -0.5 * pow((x - mean) / stddev, 2);
    return x1 * pow(M_E, x2);
}

double ProbabilityDistribution::poisson(double x, double mean) {
    assert((x - trunc(x)) == 0);
    double x1 = pow(mean, x) * pow(M_E, -mean);
    double x2 = _factorial(x);
    return x1 / x2;
}

double ProbabilityDistribution::triangular(double x, double min, double mode, double max) {
```

```
    if (x < min || x > max)
        return 0.0;
    else if (x < mode)
        return 2 * (x - min) / ((max - min)*(mode - min));
    else if (x == mode)
        return 2 / (max - min);
    else
        return 1 - pow(max - x, 2) / ((max - min)*(max - mode));
}

double ProbabilityDistribution::tStudent(double x, double mean, double stddev, double degreeFreedom) {
    double x1 = _gammaFunction((degreeFreedom + 1) / 2) / (sqrt(degreeFreedom * M_PI) * _gammaFunction(degreeFreedom / 2));
    double x2 = pow(1 + (x * x) / degreeFreedom, -(degreeFreedom + 1) / 2);
    return x1*x2;
}

double ProbabilityDistribution::uniform(double x, double min, double max) {
    return 1.0 / (max - min);
}

double ProbabilityDistribution::weibull(double x, double shape, double scale) {
    assert(shape >= 0 && scale >= 0);
    if (x > 0) {
        double x1 = (shape / scale) * pow(x / scale, shape - 1);
        double x2 = pow(M_E, -pow(x / scale, shape));
        return x1 * x2;
    } else
        return 0.0;
}
// inverse

double ProbabilityDistribution::_findInverseChi2(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double degreeFreedom) {
    double x = ((cumulativeProbability - fa) / (fb - fa)) * (b - a) + a;
    double fx = fa + integrator->integrate(a, x, ProbabilityDistribution::chi2, degreeFreedom);
    ++recursions;
    if ((recursions == integrator->getMaxSteps()) || (abs(fx - cumulativeProbability) <= integrator->getPrecision())) {
        return x;
    } else if (fx < cumulativeProbability) {
        return ProbabilityDistribution::_findInverseChi2(x, fx, b, fb, recursions, cumulativeProbability, degreeFreedom);
    } else {
        return ProbabilityDistribution::_findInverseChi2(a, fa, x, fx, recursions, cumulativeProbability, degreeFreedom);
    }
}

double ProbabilityDistribution::inverseChi2(double cumulativeProbability, double m) {
    std::string key = "chi2(" + std::to_string(m) + ")" + std::to_string(cumulativeProbability);
    auto search = ProbabilityDistribution::memory->find(key);
    if (search != ProbabilityDistribution::memory->end()) { //found
        return search->second;
    } else {
        double a, fa;
        double b, fb;
        a = 0.0;
        fa = 0.0;
        b = 1000.0;
        unsigned int savedSteps = integrator->getMaxSteps();
        integrator->setMaxSteps(1e3);
        fb = integrator->integrate(a, b, ProbabilityDistribution::chi2, m);
        //integrator->setMaxSteps(1e3);
        double inv = ProbabilityDistribution::_findInverseChi2(a, fa, b, fb, 0, cumulativeProbability, m);
        integrator->setMaxSteps(savedSteps);
        std::pair<std::string, double> pair = std::pair<std::string, double>(key, inv);
        ProbabilityDistribution::memory->insert(pair);
        return inv;
    }
}

double ProbabilityDistribution::_findInverseFFisherSnedecor(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double d1, double d2) {
    double x = ((cumulativeProbability - fa) / (fb - fa)) * (b - a) + a;
    double fx = fa + integrator->integrate(a, x, ProbabilityDistribution::fisherSnedecor, d1, d2);
    ++recursions;
    if ((recursions == integrator->getMaxSteps()) || (abs(fx - cumulativeProbability) <= integrator->getPrecision())) {
        return x;
    } else if (fx < cumulativeProbability) {
        return ProbabilityDistribution::_findInverseFFisherSnedecor(x, fx, b, fb, recursions, cumulativeProbability, d1, d2);
    } else {
        return ProbabilityDistribution::_findInverseFFisherSnedecor(a, fa, x, fx, recursions, cumulativeProbability, d1, d2);
```

```
    }
}

double ProbabilityDistribution::inverseFFisherSnedecor(double cumulativeProbability, double d1, double d2) {
    std::string key = "fisher(" + std::to_string(d1) + "," + std::to_string(d2) + ")" + std::to_string(cumulativeProbability);
    auto search = ProbabilityDistribution::memory->find(key);
    if (search != ProbabilityDistribution::memory->end()) { //found
        return search->second;
    } else {
        double a, fa;
        double b, fb;
        a = 0.0;
        fa = 0.0;
        b = 10000.0;
        unsigned int savedSteps = integrator->getMaxSteps();
        integrator->setMaxSteps(1e4);
        fb = integrator->integrate(a, b, ProbabilityDistribution::fisherSnedecor, d1, d2);
        //integrator->setMaxSteps(1e3);
        double inv = ProbabilityDistribution::_findInverseFFisherSnedecor(a, fa, b, fb, 0, cumulativeProbability, d1, d2);
        integrator->setMaxSteps(savedSteps);
        std::pair<std::string, double> pair = std::pair<std::string, double>(key, inv);
        ProbabilityDistribution::memory->insert(pair);
        return inv;
    }
}

double ProbabilityDistribution::_findInverseNormal(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double mean, double stddev) {
    double x = ((cumulativeProbability - fa) / (fb - fa)) * (b - a) + a;
    double fx = fa + integrator->integrate(a, x, ProbabilityDistribution::normal, mean, stddev);
    ++recursions;
    if ((recursions == integrator->getMaxSteps()) || (abs(fx - cumulativeProbability) <= integrator->getPrecision())) {
        return x;
    } else if (fx < cumulativeProbability) {
        return ProbabilityDistribution::_findInverseNormal(x, fx, b, fb, recursions, cumulativeProbability, mean, stddev);
    } else {
        return ProbabilityDistribution::_findInverseNormal(a, fa, x, fx, recursions, cumulativeProbability, mean, stddev);
    }
}

double ProbabilityDistribution::inverseNormal(double cumulativeProbability, double mean, double stddev) {
    std::string key = "normal(" + std::to_string(mean) + "," + std::to_string(stddev) + ")" + std::to_string(cumulativeProbability);
    auto search = ProbabilityDistribution::memory->find(key);
    if (search != ProbabilityDistribution::memory->end()) { //found
        return search->second;
    } else {
        double a, fa;
        double b, fb;
        if (cumulativeProbability <= 0.5) {
            a = mean - stddev * 5.0;
            fa = 0.0;
            b = mean;
            fb = 0.5;
        } else {
            a = mean;
            fa = 0.5;
            b = mean + stddev * 5.0;
            fb = 1.0;
        }
        double inv = ProbabilityDistribution::_findInverseNormal(a, fa, b, fb, 0, cumulativeProbability, mean, stddev);
        std::pair<std::string, double> pair = std::pair<std::string, double>(key, inv);
        ProbabilityDistribution::memory->insert(pair);
        return inv;
    }
}

double ProbabilityDistribution::_findInverseTStudent(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double mean, double stddev, double degreeFreedom) {
    double x = ((cumulativeProbability - fa) / (fb - fa)) * (b - a) + a;
    double fx = fa + integrator->integrate(a, x, ProbabilityDistribution::tStudent, mean, stddev, degreeFreedom);
    ++recursions;
    if ((recursions == integrator->getMaxSteps()) || (abs(fx - cumulativeProbability) <= integrator->getPrecision())) {
        return x;
    } else if (fx < cumulativeProbability) {
        return ProbabilityDistribution::_findInverseTStudent(x, fx, b, fb, recursions,
```

```
cumulativeProbability, mean, stddev, degreeFreedom);
    } else {
        return ProbabilityDistribution::_findInverseTStudent(a, fa, x, fx, recursions, cumulativeProbability, mean, stddev, degreeFreedom);
    }
}

double ProbabilityDistribution::inverseTStudent(double cumulativeProbability, double mean, double stddev, double degreeFreedom) {
    std::string key = "tstudent(" + std::to_string(mean) + "," + std::to_string(stddev) + "," + std::to_string(degreeFreedom) + ")" + std::to_string(cumulativeProbability);
    auto search = ProbabilityDistribution::memory->find(key);
    if (search != ProbabilityDistribution::memory->end()) { //found
        return search->second;
    } else {
        double a, fa;
        double b, fb;
        if (cumulativeProbability <= 0.5) {
            a = mean - stddev * 5.0;
            fa = 0.0;
            b = mean;
            fb = 0.5;
        } else {
            a = mean;
            fa = 0.5;
            b = mean + stddev * 5.0;
            fb = 1.0;
        }
        double inv = ProbabilityDistribution::_findInverseTStudent(a, fa, b, fb, 0, cumulativeProbability, mean, stddev, degreeFreedom);
        std::pair<std::string, double> pair = std::pair<std::string, double>(key, inv);
        ProbabilityDistribution::memory->insert(pair);
        return inv;
    }
}

//************************

double ProbabilityDistribution::_gammaFunction(double x) {
    return tgamma(x);
}

double ProbabilityDistribution::_betaFunction(double x, double y) {
    return std::beta(x, y);
}

double ProbabilityDistribution::_factorial(double x) {
    if (x > 1)
        return x * _factorial(x - 1);
    else
        return 1;
}
```

8.591 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tools/ProbabilityDistribution.h File Reference

```
#include "solver_if.h"
#include <string>
#include <map>
```

Include dependency graph for ProbabilityDistribution.h:

This graph shows which files directly or indirectly include this file:

Classes

- class ProbabilityDistribution

8.592 ProbabilityDistribution.h

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.h to edit this template
 */

/*
 * File:   ProbabilityDistribution.h
 * Author: rlcancian
 *
 * Created on 22 de novembro de 2021, 17:24
 */

#ifndef PROBABILITYDISTRIBUTION_H
#define PROBABILITYDISTRIBUTION_H

#include "solver_if.h"
#include <string>
#include <map>

class ProbabilityDistribution {
public:
    static double beta(double x, double alpha, double beta);
    static double chi2(double x, double degreeFreedom);
    static double erlang(double x, double shape, double scale); // int M
    static double exponential(double x, double mean);
    static double fisherSnedecor(double x, double d1, double d2);
    static double gamma(double x, double shape, double scale);
```

```
    static double logNormal(double x, double mean, double stddev);
    static double normal(double x, double mean, double stddev);
    static double poisson(double x, double mean);
    static double triangular(double x, double min, double mode, double max);
    static double tStudent(double x, double mean, double stddev, double degreeFreedom);
    static double uniform(double x, double min, double max);
    static double weibull(double x, double shape, double scale);
    // inverse
    static double inverseChi2(double cumulativeProbability, double degreeFreedom);
    static double inverseFFisherSnedecor(double cumulativeProbability, double d1, double d2);
    static double inverseNormal(double cumulativeProbability, double mean, double stddev);
    static double inverseTStudent(double cumulativeProbability, double mean, double stddev, double degreeFreedom);
private:
    static double _findInverseChi2(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double degreeFreedom);
    static double _findInverseFFisherSnedecor(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double d1, double d2);
    static double _findInverseNormal(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double mean, double stddev);
    static double _findInverseTStudent(double a, double fa, double b, double fb, unsigned int recursions, double cumulativeProbability, double mean, double stddev, double degreeFreedom);
private:
    static double _gammaFunction(double x);
    static double _betaFunction(double x, double y);
    static double _factorial(double x);
    //static double _inverseAnything(double x, double cumulative, double min, double max);
private:
    static std::map<std::string, double>* memory; // = std::map<std::string, double>();
    static Solver_if* integrator;
    //static IntegratorDefaultImpl1 _Integrator = IntegratorDefaultImpl1(); //Integrator_if _Integrator = Traits<Integrator_if>::Implementation(); //Traits<Integrator_if>::Precision, Traits<Integrator_if>::MaxSteps);
    //static constexpr Integrator_if* _Integrator = new Traits<Integrator_if>::Implementation(Traits<Integrator_if>::Precision, Traits<Integrator_if>::MaxSteps);
};
#endif /* PROBABILITYDISTRIBUTION_H */

```

8.593 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tools/solver_if.h File Reference

This graph shows which files directly or indirectly include this file:

Classes

- class Solver_if

8.594 solver_if.h

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   Integrator_if.h
```

```
 * Author: rafael.luiz.cancian
 *
 * Created on 14 de Agosto de 2018, 13:54
 */

#ifndef SOLVER_IF_H
#define SOLVER_IF_H

class Solver_if {
public:
    virtual void setPrecision(double e) = 0;
    virtual double getPrecision() = 0;
    virtual void setMaxSteps(double steps) = 0;
    virtual double getMaxSteps() = 0;
    virtual double integrate(double min, double max, double (*f)(double, double), double p2) = 0;
    virtual double integrate(double min, double max, double (*f)(double, double, double), double p2, double p3) = 0;
    virtual double integrate(double min, double max, double (*f)(double, double, double, double), double p2, double p3, double p4) = 0;
    virtual double integrate(double min, double max, double (*f)(double, double, double, double, double), double p2, double p3, double p4, double p5) = 0;
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double), double p2) = 0;
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double, double), double p2, double p3) = 0;
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double, double, double), double p2, double p3, double p4) = 0;
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double, double, double, double), double p2, double p3, double p4, double p5) = 0;
};

#endif /* SOLVER_IF_H */

```

8.595 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/tools/SolverDefaultImpl1.cpp File Reference

```
#include "SolverDefaultImpl1.h"
#include <math.h>
```

Include dependency graph for SolverDefaultImpl1.cpp:

8.596 SolverDefaultImpl1.cpp

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.cc to edit this template
 */

/*
 * File:   IntegratorDefaultImpl1.cpp
 * Author: rlcancian
 *
 * Created on 23 de novembro de 2021, 18:36
 */

#include "SolverDefaultImpl1.h"
#include <math.h>

SolverDefaultImpl1::SolverDefaultImpl1(double precision, unsigned int steps) {
    _precision = precision;
    _numSteps = steps;
}

void SolverDefaultImpl1::setPrecision(double precision) {
    _precision = precision;
}

double SolverDefaultImpl1::getPrecision() {
    return _precision;
}

void SolverDefaultImpl1::setMaxSteps(double steps) {
    _numSteps = steps;
}

double SolverDefaultImpl1::getMaxSteps() {
    return _numSteps;
}

double SolverDefaultImpl1::integrate(double min, double max, double (*f)(double, double), double p2) {
    // Simpson's 1/3 rule
    double steps = _numSteps;
    double h = (max - min) / steps; // distance between points
    if (h < _precision) {
        steps = (max - min) / _precision;
        h = (max - min) / steps;
    }
    double x, c; // x is the point being evaluated, c is the weight
    double sum = 0.0;
    for (int i = 0; i <= steps; i++) {
        if (i == 0 || i == steps)
            c = 1;
        else if (i % 2 == 0)
            c = 2;
        else
            c = 4;
        x = min + i*h;
        sum += c * f(x, p2);
    }
    return (h / 3)*sum;
}

double SolverDefaultImpl1::integrate(double min, double max, double (*f)(double, double, double), double p2, double p3) {
    // Simpson's 1/3 rule

    unsigned int steps = _numSteps;
    double h = (max - min) / steps; // distance between points
    if (h < _precision) {
        steps = ceil((max - min) / _precision);
        h = (max - min) / steps;
    }
    double x, c; // x is the point being evaluated, c is the weight
    double sum = 0.0;
    for (int i = 0; i <= steps; i++) {
        if (i == 0 || i == steps)
            c = 1;
        else if (i % 2 == 0)
            c = 2;
        else
            c = 4;
        x = min + i*h;
        sum += c * f(x, p2, p3);
    }
    return (h / 3)*sum;
}

double SolverDefaultImpl1::integrate(double min, double max, double (*f)(double, double, double,
```

```
double), double p2, double p3, double p4) {
    // Simpson's 1/3 rule
    double steps = _numSteps;
    double h = (max - min) / steps; // distance between points
    if (h < _precision) {
        steps = (max - min) / _precision;
        h = (max - min) / steps;
    }
    double x, c; // x is the point being evaluated, c is the weight
    double sum = 0.0;
    for (int i = 0; i <= steps; i++) {
        if (i == 0 || i == steps)
            c = 1;
        else if (i % 2 == 0)
            c = 2;
        else
            c = 4;
        x = min + i*h;
        sum += c * f(x, p2, p3, p4);
    }
    return (h / 3)*sum;
}

double SolverDefaultImpl1::integrate(double min, double max, double (*f)(double, double, double,
 double, double), double p2, double p3, double p4, double p5) {
    // Simpson's 1/3 rule
    double steps = _numSteps;
    double h = (max - min) / steps; // distance between points
    if (h < _precision) {
        steps = (max - min) / _precision;
        h = (max - min) / steps;
    }
    double x, c; // x is the point being evaluated, c is the weight
    double sum = 0.0;
    for (int i = 0; i <= steps; i++) {
        if (i == 0 || i == steps)
            c = 1;
        else if (i % 2 == 0)
            c = 2;
        else
            c = 4;
        x = min + i*h;
        sum += c * f(x, p2, p3, p4, p5);
    }
    return (h / 3)*sum;
}

double SolverDefaultImpl1::derivate(double initPoint, double initValue, double (*f)(double, double),
 double p2) {
    double time, halfStep;
    unsigned int i, numEqs = 1;
    double k1[numEqs], k2[numEqs], k3[numEqs], k4[numEqs], valVar[numEqs], result[numEqs];
    time = initPoint;
    halfStep = _stepSize * 0.5;
    for (i = 0; i < numEqs; i++) {
        k1[i] = f(time, p2);
    }
    time += halfStep;
    for (i = 0; i < numEqs; i++) {
        k2[i] = f(time, p2 + k1[i] * halfStep);
    }
    for (i = 0; i < numEqs; i++) {
        k3[i] = f(time, p2 + k2[i] * halfStep);
    }
    for (i = 0; i < numEqs; i++) {
        k4[i] = f(time, p2 + k3[i] * halfStep);
    }
    for (i = 0; i < numEqs; i++) {
        result[i] = p2 + (_stepSize / 6) * (k1[i] + 2 * (k2[i] + k3[i]) + k4[i]);
    }
    time = initPoint + _stepSize; // OR  time += halfStep;
    return result[0];

}

double SolverDefaultImpl1::derivate(double initPoint, double initValue, double (*f)(double, double,
 double), double p2, double p3) {
}

double SolverDefaultImpl1::derivate(double initPoint, double initValue, double (*f)(double, double,
 double, double), double p2, double p3, double p4) {
}

double SolverDefaultImpl1::derivate(double initPoint, double initValue, double (*f)(double, double,
 double, double, double), double p2, double p3, double p4, double p5) {
}
```

8.597 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/tools/SolverDefaultImpl1.h File Reference

`#include "solver_if.h"`
Include dependency graph for SolverDefaultImpl1.h:

This graph shows which files directly or indirectly include this file:

Classes

- class SolverDefaultImpl1

8.598 SolverDefaultImpl1.h

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/cppFiles/class.h to edit this template
 */

/*
 * File:   IntegratorDefaultImpl1.h
 * Author: rlcancian
 *
 * Created on 23 de novembro de 2021, 18:36
 */

#ifndef SOLVERDEFAULTIMPL1_IF_H
#define SOLVERDEFAULTIMPL1_IF_H

#include "solver_if.h"
```

```
class SolverDefaultImpl1 : public Solver_if {
public:
    SolverDefaultImpl1(double precision = 1e-6, unsigned int steps = 1e3);
    virtual ~SolverDefaultImpl1() = default;
public:
    virtual void setPrecision(double e);
    virtual double getPrecision();
    virtual void setMaxSteps(double steps);
    virtual double getMaxSteps();
    virtual double integrate(double min, double max, double (*f)(double, double), double p2);
    virtual double integrate(double min, double max, double (*f)(double, double, double), double p2, double p3);
    virtual double integrate(double min, double max, double (*f)(double, double, double, double), double p2, double p3, double p4);
    virtual double integrate(double min, double max, double (*f)(double, double, double, double, double), double p2, double p3, double p4, double p5);
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double), double p2);
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double, double), double p2, double p3);
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double, double, double), double p2, double p3, double p4);
    virtual double derivate(double initPoint, double initValue, double (*f)(double, double, double, double, double), double p2, double p3, double p4, double p5);
private:
    double _precision;
    unsigned int _numSteps;
    double _stepSize;
    //Variable* _variables;
    //Variable* _timeVariable;
};

#endif /* SOLVERDEFAULTIMPL1_IF_H */
```

8.599 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/Traits.h File Reference

```
#include "kernel/util/Util.h"
#include "applications/terminal/GenesysTerminalApp.h"
#include "applications/gui/qt/GenesysQtGUI.h"
#include "examples/book/Book_Cap02_Example01.h"
#include "examples/smarts/Smart_AssignWriteSeizes.h"
#include "examples/smarts/Smart_Delay.h"
#include "examples/smarts/Smart_Dummy.h"
#include "examples/smarts/Smart_HoldSignal.h"
#include "examples/smarts/Smart_ModelInfoModelSimulation.h"
#include "examples/smarts/Smart_OnEvent.h"
#include "examples/smarts/Smart_Parser.h"
#include "examples/smarts/Smart_ParserModelFunctions.h"
#include "examples/smarts/Smart_Process.h"
#include "examples/smarts/Smart_ProcessSet.h"
#include "examples/smarts/Smart_RouteStation.h"
#include "examples/smarts/Smart_SeizeDelayRelease.h"
#include "examples/smarts/Smart_SeizeDelayReleaseMany.h"
#include "examples/smarts/Smart_Sequence.h"
#include "examples/teaching/AnElectronicAssemblyAndTestSystem.h"
#include "examples/teaching/OperatingSystem02.h"
#include "examples/teaching/OperatingSystem03.h"
#include "tools/SolverDefaultImpl1.h"
#include "tools/HypothesisTesterDefaultImpl1.h"
#include "tools/HypothesisTester_if.h"
```

This graph shows which files directly or indirectly include this file:

Classes

- struct Traits< T >
- struct Traits< GenesysApplication_if >
- struct Traits< Solver_if >
- struct Traits< HypothesisTester_if >

8.600 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/Traits.h

```
00001 /*
00002  * To change this license header, choose License Headers in Project Properties.
00003  * To change this template file, choose Tools | Templates
00004  * and open the template in the editor.
00005  */
00006
00007 /*
00008  * File:   Traits.h
00009  * Author: rafael.luiz.cancian
00010  *
00011  * Created on 14 de Agosto de 2018, 19:36
00012  */
00013
00014 #ifndef TRAITS_H
00015 #define TRAITS_H
00016
00017 #include "kernel/util/Util.h"
00018 // USER INTERFACES
00019 #include "applications/terminal/GenesysTerminalApp.h"
00020 #include "applications/gui/qt/GenesysQtGUI.h"
00021 //
00022 // EXAMPLES
00023 //
00024 // book
00025 #include "examples/book/Book_Cap02_Example01.h"
00026 // smarts
00027 #include "examples/smarts/Smart_AssignWriteSeizes.h"
00028 #include "examples/smarts/Smart_Delay.h"
00029 #include "examples/smarts/Smart_Dummy.h"
00030 #include "examples/smarts/Smart_HoldSignal.h"
00031 #include "examples/smarts/Smart_ModelInfoModelSimulation.h"
00032 #include "examples/smarts/Smart_OnEvent.h"
00033 #include "examples/smarts/Smart_Parser.h"
00034 #include "examples/smarts/Smart_ParserModelFunctions.h"
00035 #include "examples/smarts/Smart_Process.h"
00036 #include "examples/smarts/Smart_ProcessSet.h"
00037 #include "examples/smarts/Smart_RouteStation.h"
00038 #include "examples/smarts/Smart_SeizeDelayRelease.h"
00039 #include "examples/smarts/Smart_SeizeDelayReleaseMany.h"
00040 #include "examples/smarts/Smart_Sequence.h"
00041 // teachng
00042 #include "examples/teaching/AnElectronicAssemblyAndTestSystem.h"
00043 #include "examples/teaching/OperatingSystem02.h"
00044 #include "examples/teaching/OperatingSystem03.h"
00045 //
00046 // TOOLS
00047 #include "tools/SolverDefaultImpl1.h"
00048 #include "tools/HypothesisTesterDefaultImpl1.h"
00049 #include "tools/HypothesisTester_if.h"
00050
00051 template <typename T>
00052 struct Traits {
00053 };
00054
```

```
template <> struct Traits<GenesysApplication_if> {
    static const Util::TraceLevel traceLevel = Util::TraceLevel::L6_arrival;

    //typedef Smart_AssignWriteSeizes Application;
    //typedef Smart_Delay Application;
    //typedef Smart_Dummy Application;
    //typedef Smart_HoldSignal Application;
    //typedef Smart_ModelInfoModelSimulation Application;
    typedef Smart_OnEvent Application;
    //typedef Smart_Parser Application;
    //typedef Smart_ParserModelFunctions Application;
    //typedef Smart_Process Application;
    //typedef Smart_ProcessSet Application;
    //typedef Smart_RouteStation Application;
    //typedef Smart_SeizeDelayRelease Application;
    //typedef Smart_SeizeDelayReleaseMany Application;
    //typedef Smart_Sequence Application;

    //typedef AnElectronicAssemblyAndTestSystem Application;
    //typedef OperatingSystem02 Application;
    //typedef OperatingSystem03 Application;

    //typedef GenesysTerminalApp Application;
    //typedef GenesysQtGUI Application;

};

template <> struct Traits<Solver_if> {
    typedef SolverDefaultImpl1 Implementation;
    static constexpr double Precision = 1e-5;
    static constexpr unsigned int MaxSteps = 1e2;
};

template <> struct Traits<HypothesisTester_if> {
    typedef HypothesisTesterDefaultImpl1 Implementation;
    static constexpr unsigned int ConfidenceLevel = 95;
};
#endif /* TRAITS_H */
```

8.601 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/ui_dialogabout.h File Reference

```
#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QLabel>
#include <QtWidgets/QTextEdit>
#include <QtWidgets/QVBoxLayout>
```

Include dependency graph for ui_dialogabout.h:

Classes

- class Ui_DialogAbout
- class Ui::DialogAbout

Namespaces

- Ui

8.602 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/ui_dialogabout.h

```
/********************************************************************************
** Form generated from reading UI file 'dialogabout.ui'
**
** Created by: Qt User Interface Compiler version 5.15.2
**
** WARNING! All changes made in this file will be lost when recompiling UI file!
********************************************************************************/

#ifndef UI_DIALOGABOUT_H
#define UI_DIALOGABOUT_H

#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QLabel>
#include <QtWidgets/QTextEdit>
#include <QtWidgets/QVBoxLayout>

QT_BEGIN_NAMESPACE

class Ui_DialogAbout
{
public:
    QVBoxLayout *verticalLayout_3;
    QHBoxLayout *horizontalLayout_2;
    QLabel *labelIcon;
    QVBoxLayout *verticalLayout_2;
    QLabel *label_2;
    QLabel *labelAuthor;
    QLabel *labelInstitution;
    QLabel *labelVersionGenesys;
    QTextEdit *textEditDescription;
    QDialogButtonBox *buttonOk;

    void setupUi(QDialog *DialogAbout)
    {
        if (DialogAbout->objectName().isEmpty())
            DialogAbout->setObjectName(QString::fromUtf8("DialogAbout"));
        DialogAbout->resize(469, 227);
        DialogAbout->setLocale(QLocale(QLocale::English, QLocale::UnitedStates));
        DialogAbout->setModal(true);
        verticalLayout_3 = new QVBoxLayout(DialogAbout);
        verticalLayout_3->setObjectName(QString::fromUtf8("verticalLayout_3"));
        horizontalLayout_2 = new QHBoxLayout();
        horizontalLayout_2->setObjectName(QString::fromUtf8("horizontalLayout_2"));
        labelIcon = new QLabel(DialogAbout);
        labelIcon->setObjectName(QString::fromUtf8("labelIcon"));
        labelIcon->setPixmap(QPixmap(QString::fromUtf8(":/rec/resources/software_genesys.gif")));

        horizontalLayout_2->addWidget(labelIcon);

        verticalLayout_2 = new QVBoxLayout();
        verticalLayout_2->setObjectName(QString::fromUtf8("verticalLayout_2"));
        label_2 = new QLabel(DialogAbout);
        label_2->setObjectName(QString::fromUtf8("label_2"));
        QFont font;
        font.setPointSize(12);
        font.setBold(true);
        font.setWeight(75);
        label_2->setFont(font);
        label_2->setFrameShadow(QFrame::Raised);
        label_2->setTextFormat(Qt::RichText);

        verticalLayout_2->addWidget(label_2);

        labelAuthor = new QLabel(DialogAbout);
        labelAuthor->setObjectName(QString::fromUtf8("labelAuthor"));

        verticalLayout_2->addWidget(labelAuthor);

        labelInstitution = new QLabel(DialogAbout);
        labelInstitution->setObjectName(QString::fromUtf8("labelInstitution"));

        verticalLayout_2->addWidget(labelInstitution);

        labelVersionGenesys = new QLabel(DialogAbout);
        labelVersionGenesys->setObjectName(QString::fromUtf8("labelVersionGenesys"));

        verticalLayout_2->addWidget(labelVersionGenesys);

```

```
        horizontalLayout_2->addLayout(verticalLayout_2);


        verticalLayout_3->addLayout(horizontalLayout_2);

        textEditDescription = new QTextEdit(DialogAbout);
        textEditDescription->setObjectName(QString::fromUtf8("textEditDescription"));
        QFont font1;
        font1.setPointSize(9);
        textEditDescription->setFont(font1);
        textEditDescription->setFrameShape(QFrame::NoFrame);
        textEditDescription->setFrameShadow(QFrame::Plain);
        textEditDescription->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
        textEditDescription->setReadOnly(true);

        verticalLayout_3->addWidget(textEditDescription);

        buttonOk = new QDialogButtonBox(DialogAbout);
        buttonOk->setObjectName(QString::fromUtf8("buttonOk"));
        buttonOk->setOrientation(Qt::Horizontal);
        buttonOk->setStandardButtons(QDialogButtonBox::Ok);

        verticalLayout_3->addWidget(buttonOk);


        retranslateUi(DialogAbout);
        QObject::connect(buttonOk, SIGNAL(accepted()), DialogAbout, SLOT(accept()));

        QMetaObject::connectSlotsByName(DialogAbout);
    } // setupUi

    void retranslateUi(QDialog *DialogAbout)
    {
        DialogAbout->setWindowTitle(QCoreApplication::translate("DialogAbout", "About GenESyS", nullptr));
        labelIcon->setText(QString());
        label_2->setText(QCoreApplication::translate("DialogAbout", "Generic and Expansible System Simulator", nullptr));
        labelAuthor->setText(QCoreApplication::translate("DialogAbout", "Author: Prof. Dr. Ing. Rafael Luiz Cancian", nullptr));
        labelInstitution->setText(QCoreApplication::translate("DialogAbout", "Federal University of Santa Catarina, Brazil", nullptr));
        labelVersionGenesys->setText(QCoreApplication::translate("DialogAbout", "Version 19.11 (DayOfTheDead 19)", nullptr));
        textEditDescription->setHtml(QCoreApplication::translate("DialogAbout", "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0//EN\" \"http://www.w3.org/TR/REC-html40/strict.dtd\">\n"
"<html><head><meta name=\"qrichtext\" content=\"1\" /><style type=\"text/css\">\n"
"p, li { white-space: pre-wrap; }\n"
"</style></head><body style=\" font-family:'DejaVu LGC Sans'; font-size:9pt; font-weight:400; font-style:normal;\" bgcolor=\"#eeeeee\">\n"
"<p style=\" margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-indent:0px;\"><span style=\" font-family:'Tahoma,Arial,sans-serif'; font-size:8pt; color:#000000;\">Genesys is a result of teaching and researchactivities of Professor Dr. Ing Rafael Luiz Cancian. It began in early 2002 as a way to teach students the basics andsimulation techniques of systems implemented by other comercial simulation tools, such as Arena. In Genesysdevelopment he replicated all the SIMAN language, used by Arena software, and Genesys has become a clone of thattool, including its graphical interface. Genesys allowed the inclusion of ne"
                                "w simulation components through dynamiclink libraries and also the parallel execution of simulation models in a distributed environment. The developmentof Genesys continued until 2007, when the professor stopped teaching systems simulation.</span></p></body></html>", nullptr));
    } // retranslateUi

};

namespace Ui {
    class DialogAbout: public Ui_DialogAbout {};
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_DIALOGABOUT_H
```

8.603 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_dialogmodelinformation.h File Reference

```
#include <QtCore/QLocale>
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QApplication>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QLabel>
#include <QtWidgets/QLineEdit>
#include <QtWidgets/QVBoxLayout>
```

Include dependency graph for ui_dialogmodelinformation.h:

Classes

- class Ui_DialogModelInformation
- class Ui::DialogModelInformation

Namespaces

- Ui

8.604 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/ui_dialogmodelinformation.h

```
/********************************************************************************
** Form generated from reading UI file 'dialogmodelinformation.ui'
**
** Created by: Qt User Interface Compiler version 5.15.2
**
** WARNING! All changes made in this file will be lost when recompiling UI file!
********************************************************************************/

#ifndef UI_DIALOGMODELINFORMATION_H
#define UI_DIALOGMODELINFORMATION_H

#include <QtCore/QLocale>
#include <QtCore/QVariant>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QLabel>
#include <QtWidgets/QLineEdit>
#include <QtWidgets/QVBoxLayout>

QT_BEGIN_NAMESPACE

class Ui_DialogModelInformation
{
public:
    QVBoxLayout *verticalLayout;
    QHBoxLayout *horizontalLayout_2;
    QLabel *label_2;
    QLineEdit *lineEdit_AnalystName;
    QHBoxLayout *horizontalLayout_3;
    QLabel *label_4;
    QLineEdit *lineEdit_ProjectTitle;
    QHBoxLayout *horizontalLayout;
    QLabel *label;
    QLineEdit *lineEdit_ModelName;
    QHBoxLayout *horizontalLayout_4;
    QLabel *label_3;
    QLineEdit *lineEdit_Description;
    QHBoxLayout *horizontalLayout_5;
    QLabel *label_5;
    QLineEdit *lineEdit_Version;
```

```
    QDialogButtonBox *buttonBox;

    void setupUi(QDialog *DialogModelInformation)
    {
        if (DialogModelInformation->objectName().isEmpty())
            DialogModelInformation->setObjectName(QString::fromUtf8("DialogModelInformation"));
        DialogModelInformation->resize(400, 300);
        DialogModelInformation->setLocale(QLocale(QLocale::English, QLocale::UnitedStates));
        verticalLayout = new QVBoxLayout(DialogModelInformation);
        verticalLayout->setObjectName(QString::fromUtf8("verticalLayout"));
        horizontalLayout_2 = new QHBoxLayout();
        horizontalLayout_2->setObjectName(QString::fromUtf8("horizontalLayout_2"));
        label_2 = new QLabel(DialogModelInformation);
        label_2->setObjectName(QString::fromUtf8("label_2"));

        horizontalLayout_2->addWidget(label_2);

        lineEdit_AnalystName = new QLineEdit(DialogModelInformation);
        lineEdit_AnalystName->setObjectName(QString::fromUtf8("lineEdit_AnalystName"));

        horizontalLayout_2->addWidget(lineEdit_AnalystName);


        verticalLayout->addLayout(horizontalLayout_2);

        horizontalLayout_3 = new QHBoxLayout();
        horizontalLayout_3->setObjectName(QString::fromUtf8("horizontalLayout_3"));
        label_4 = new QLabel(DialogModelInformation);
        label_4->setObjectName(QString::fromUtf8("label_4"));

        horizontalLayout_3->addWidget(label_4);

        lineEdit_ProjectTitle = new QLineEdit(DialogModelInformation);
        lineEdit_ProjectTitle->setObjectName(QString::fromUtf8("lineEdit_ProjectTitle"));

        horizontalLayout_3->addWidget(lineEdit_ProjectTitle);


        verticalLayout->addLayout(horizontalLayout_3);

        horizontalLayout = new QHBoxLayout();
        horizontalLayout->setObjectName(QString::fromUtf8("horizontalLayout"));
        label = new QLabel(DialogModelInformation);
        label->setObjectName(QString::fromUtf8("label"));

        horizontalLayout->addWidget(label);

        lineEdit_ModelName = new QLineEdit(DialogModelInformation);
        lineEdit_ModelName->setObjectName(QString::fromUtf8("lineEdit_ModelName"));

        horizontalLayout->addWidget(lineEdit_ModelName);


        verticalLayout->addLayout(horizontalLayout);

        horizontalLayout_4 = new QHBoxLayout();
        horizontalLayout_4->setObjectName(QString::fromUtf8("horizontalLayout_4"));
        label_3 = new QLabel(DialogModelInformation);
        label_3->setObjectName(QString::fromUtf8("label_3"));

        horizontalLayout_4->addWidget(label_3);

        lineEdit_Description = new QLineEdit(DialogModelInformation);
        lineEdit_Description->setObjectName(QString::fromUtf8("lineEdit_Description"));

        horizontalLayout_4->addWidget(lineEdit_Description);


        verticalLayout->addLayout(horizontalLayout_4);

        horizontalLayout_5 = new QHBoxLayout();
        horizontalLayout_5->setObjectName(QString::fromUtf8("horizontalLayout_5"));
        label_5 = new QLabel(DialogModelInformation);
        label_5->setObjectName(QString::fromUtf8("label_5"));

        horizontalLayout_5->addWidget(label_5);

        lineEdit_Version = new QLineEdit(DialogModelInformation);
        lineEdit_Version->setObjectName(QString::fromUtf8("lineEdit_Version"));

        horizontalLayout_5->addWidget(lineEdit_Version);


        verticalLayout->addLayout(horizontalLayout_5);

        buttonBox = new QDialogButtonBox(DialogModelInformation);
        buttonBox->setObjectName(QString::fromUtf8("buttonBox"));
```

```
        buttonBox->setOrientation(Qt::Horizontal);
        buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);

        verticalLayout->addWidget(buttonBox);

#if QT_CONFIG(shortcut)
        label_2->setBuddy(lineEdit_AnalystName);
        label_4->setBuddy(lineEdit_ProjectTitle);
        label->setBuddy(lineEdit_ModelName);
        label_3->setBuddy(lineEdit_Description);
        label_5->setBuddy(lineEdit_Version);
#endif // QT_CONFIG(shortcut)
        QWidget::setTabOrder(lineEdit_AnalystName, lineEdit_ProjectTitle);
        QWidget::setTabOrder(lineEdit_ProjectTitle, lineEdit_ModelName);
        QWidget::setTabOrder(lineEdit_ModelName, lineEdit_Description);
        QWidget::setTabOrder(lineEdit_Description, lineEdit_Version);

        retranslateUi(DialogModelInformation);
        QObject::connect(buttonBox, SIGNAL(accepted()), DialogModelInformation, SLOT(accept()));
        QObject::connect(buttonBox, SIGNAL(rejected()), DialogModelInformation, SLOT(reject()));

        QMetaObject::connectSlotsByName(DialogModelInformation);
    } // setupUi

    void retranslateUi(QDialog *DialogModelInformation)
    {
        DialogModelInformation->setWindowTitle(QCoreApplication::translate("DialogModelInformation",
 "Model Information", nullptr));
        label_2->setText(QCoreApplication::translate("DialogModelInformation", "Analyst Name:",
 nullptr));
        label_4->setText(QCoreApplication::translate("DialogModelInformation", "Project Title:",
 nullptr));
        label->setText(QCoreApplication::translate("DialogModelInformation", "Model Name:", nullptr));
        label_3->setText(QCoreApplication::translate("DialogModelInformation", "Description:",
 nullptr));
        label_5->setText(QCoreApplication::translate("DialogModelInformation", "Version:", nullptr));
    } // retranslateUi

};

namespace Ui {
    class DialogModelInformation: public Ui_DialogModelInformation {};
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_DIALOGMODELINFORMATION_H
```

8.605 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/↩ RebornedGenESyS/2019_2022_GenESyS/ui_formPlugins.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
```

Include dependency graph for ui_formPlugins.h:

Classes

- class Ui_formPlugins
- class Ui::formPlugins

Namespaces

- Ui

8.606 /home/rlcancian/Laboratory/Software_Lab/IA32_Architecture/GccProjects/RebornedGenESyS/2019_2022_GenESyS/ui_formPlugins.h

```
/********************************************************************************
** Form generated from reading UI file 'formPlugins.ui'
**
** Created by: Qt User Interface Compiler version 5.15.2
**
** WARNING! All changes made in this file will be lost when recompiling UI file!
********************************************************************************/

#ifndef UI_FORMPLUGINS_H
#define UI_FORMPLUGINS_H

#include <QtCore/QVariant>
#include <QtWidgets/QApplication>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>

QT_BEGIN_NAMESPACE

class Ui_formPlugins
{
public:
    QDialogButtonBox *buttonBox;

    void setupUi(QDialog *formPlugins)
    {
        if (formPlugins->objectName().isEmpty())
            formPlugins->setObjectName(QString::fromUtf8("formPlugins"));
        formPlugins->resize(632, 465);
        buttonBox = new QDialogButtonBox(formPlugins);
        buttonBox->setObjectName(QString::fromUtf8("buttonBox"));
        buttonBox->setGeometry(QRect(270, 390, 341, 32));
        buttonBox->setOrientation(Qt::Horizontal);
        buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);

        retranslateUi(formPlugins);
        QObject::connect(buttonBox, SIGNAL(accepted()), formPlugins, SLOT(accept()));
        QObject::connect(buttonBox, SIGNAL(rejected()), formPlugins, SLOT(reject()));

        QMetaObject::connectSlotsByName(formPlugins);
    } // setupUi

    void retranslateUi(QDialog *formPlugins)
    {
        formPlugins->setWindowTitle(QCoreApplication::translate("formPlugins", "Plugins", nullptr));
    } // retranslateUi

};

namespace Ui {
    class formPlugins: public Ui_formPlugins {};
} // namespace Ui

QT_END_NAMESPACE

#endif // UI_FORMPLUGINS_H
```