Concept Library Model Documentation

Copyright (c) 2018 Model Driven Solutions, Inc.

Contents

1 Concept Library 19
1.1.1 Diagram: Federated Concept Library 19
1.1.2 Diagram: Foundation 20
1.2 Concept Library::Abilities 22
1.2.1 Diagram: Ability 22
1.2.2 Diagram: Alter ability or control 23
1.2.3 Association Class Ability <<Relationship>> 23
1.2.4 Association Class Affected Available Resource <<Relationship>> 24
1.2.5 Class Alter Capability 25
1.2.6 Class Facilitator <<Role>> 25
1.2.7 Class Lose Ability 25
1.2.8 Class Obtain Ability 25
1.2.9 Association Class Strenthened Actor <<Relationship>> 25
1.2.10 Class Transfer Ability 26
1.2.11 Association Class Weakened Actor <<Relationship>> 26
1.3 Concept Library::Assertions 28
1.3.1 Diagram: Assertions 28
1.4 Concept Library::Assessments 29
1.4.1 Diagram: Assessment 29
1.4.2 Class Assessed Entity <<Role>> 29
1.4.3 Class Assessment 29
1.4.4 Class Assessment Activity 30
1.4.5 Association Class Entity Assessment <<Relationship>> 30
1.5 Concept Library::Contact Information 31
1.5.1 Diagram: Contact Information 31
1.5.2 Class Communications Security Level <<Value>> 32
1.5.3 Class Contact Means <<Value>> 32
1.5.4 Association Class Contact Via <<Relationship>> 32
1.5.5 Class Contactable 33
1.5.6 Class Electronic Contact <<Value>> 33
1.5.7 Class Email Address <<Value>> 34
1.5.8 Class Internet Contact <<Value>> 34
1.5.9 Class Network Identifier <<Value>> 34
1.5.10 Class Postal Address <<Value>> 34
1.5.11 Class Postal Address Structured <<Value>> 34
1.5.12 Class Postal Address Text <<Value>> 35
1.5.13 Class Postal Code <<Value>> 35
1.5.14 Class Private Network Contact <<Value>> 36
1.5.15 Class Radio Contact <<Value>> 36

1.5.16 Class Social Network Contact <<Value>> 36
1.5.17 Class Telephone Area Code <<Value>> 36
1.5.18 Class Telephone Country Code <<Value>> 36
1.5.19 Class Telephone Number <<Value>> 37
1.5.20 Class Telephone Number Structured <<Value>> 37
1.5.21 Class Telephone Number Text <<Value>> 37
1.5.22 Class Website Contact <<Value>> 37
1.6 Concept Library::Containment 40
1.6.1 Diagram: Containment 40
1.6.2 Diagram: Move Between Containers 41
1.6.3 Diagram: Physical Containment 42
1.6.4 Class Add To Container Event 42
1.6.5 Class Container <<Role>> 42
1.6.6 Association Class Containment <<Relationship>> 42
1.6.7 Class Containment Event 43
1.6.8 Class Physical Container <<Role>> 43
1.6.9 Association Class Physical Containment <<Relationship>> 43
1.6.10 Association Class Recieving Container <<Relationship>> 44
1.6.11 Class Relocation 44
1.6.12 Class Removal Event 44
1.6.13 Association Class Supplying Container <<Relationship>> 45
1.7 Concept Library::Control 46
1.7.1 Diagram: Control 46
1.7.2 Diagram: Control Authority 47
1.7.3 Diagram: Custody 47
1.7.4 Class Authority <<Role>> 48
1.7.5 Association Class Control <<Relationship>> 48
1.7.6 Class Controlled Entity <<Role>> 48
1.7.7 Class Controlling Actor <<Role>> 49
1.7.8 Class Custodian <<Role>> 49
1.7.9 Association Class Custody <<Relationship>> 49
1.7.10 Class Leader <<Role>> 49
1.7.11 Association Class Leadership <<Relationship>> 49
1.7.12 Class Managed Entity <<Role>> 50
1.7.13 Class Owner <<Role>> 50
1.7.14 Association Class Ownership Control <<Relationship>> 50
1.7.15 Association Class Possession <<Relationship>> 50
1.7.16 Class Property <<Role>> 50
1.7.17 Association Class Subject to Authority <<Relationship>> 51
1.8 Concept Library::Credentials 52
1.8.1 Diagram: Credentials and Managed Identifiers 52
1.8.2 Association Actor Identifier of Credential 52
1.8.3 Association Class Attest to Ability <<Relationship>> 53
1.8.4 Class Credential 53
1.8.5 Class Identity Provider <<Role>> 54
1.8.6 Association Class Issue Credential <<Relationship>> 54
1.8.7 Class Managed Actor Identifier <<Value>> 54
1.8.8 Association Class Valid for Time Interval <<Relationship>> 55
1.9 Concept Library::Cyber 56
1.9.1 Diagram: Cyber 57
1.9.2 Diagram: Cyber Platforms 58
1.9.3 Diagram: Cyber Resource 59
1.9.4 Association Class Ability To Execute Software <<Relationship>> 59
1.9.5 Association Class Automated Capability <<Relationship>> 60
1.9.6 Association Class Automated Control <<Relationship>> 61

1.9.7 Class Automated Entity <<Role>> 61
1.9.8 Class Automation Type 62
1.9.9 Class Automaton 62
1.9.10 Class Communicating Device 62
1.9.11 Class Communications Link 62
1.9.12 Class Communications Network 62
1.9.13 Class Computer Control System <<Role>> 62
1.9.14 Class Computer System 63
1.9.15 Class Cyber Resource <<Union>> 63
1.9.16 Class Execution Platform <<Union>> 63
1.9.17 Association Class Information In Computer <<Relationship>> 63
1.9.18 Association Class Node of a Network <<Relationship>> 64
1.9.19 Class Software 64
1.10 Concept Library::Enterprises 65
1.10.1 Diagram: Enterprise 65
1.10.2 Class Enterprise 66
1.11 Concept Library::Entities 67
1.11.1 Diagram: Identifiable Entity 68
1.11.2 Diagram: Identifiable Entity Relationships 69
1.11.3 Association Class Impact <<Relationship>> 69
1.11.4 Association Class Temporally Related <<Relationship>> 70
1.12 Concept Library::Identifiers and Coordinates 72
1.12.1 Diagram: Identifiers 73
1.12.2 Class Coordinate <<Value>> 73
1.12.3 Class Coordinate System 73
1.12.4 Association System of Coordinate 74
1.13 Concept Library::Information 75
1.13.1 Diagram: Information Action 75
1.13.2 Diagram: Information Objects 76
1.13.3 Diagram: Metadata 77
1.13.4 Class Add Information 77
1.13.5 Class Atomic Information Object 77
1.13.6 Class Close Information 78
1.13.7 Class Confidence 78
1.13.8 Association Confidence in Assertion 78
1.13.9 Association Class Contained Information <<Relationship>> 78
1.13.10 Class Copy Information 79
1.13.11 Class Create Information 79
1.13.12 Class Delete Information 79
1.13.13 Class Document 79
1.13.14 Class Information Action 80
1.13.15 Class Information Object 80
1.13.16 Class Information Repository 80
1.13.17 Class Information Resource <<Union>> 80
1.13.18 Class Information Type 81
1.13.19 Class Modify Information 81
1.13.20 Class Open Information 81
1.13.21 Class Read Information 81
1.13.22 Class Remove Information 81
1.13.23 Class Structured Information Object 81
1.13.24 Class Transfer Information 82
1.14 Concept Library::Locations 83
1.14.1 Diagram: Location 83
1.14.2 Diagram: Location Identification 84
1.14.3 Association Address of Location 84

1.14.4 Association Coordinate of location....85
1.14.5 Association Designation of a Location....85
1.14.6 Class Location ID <<Value>>....85
1.14.7 Class Location Identifier <<Value>>....85
1.14.8 Association Class Physical Boundary <<Relationship>>....85
1.14.9 Class Physical Location....86
1.14.10 Class Physical Point....86
1.14.11 Class Point On Earth <<Value>>....86
1.14.12 Association Reference Point....87
1.14.13 Class Relative Coordinate <<Value>>....87
1.14.14 Class Spacial Coordinate <<Value>>....87
1.14.15 Association Topological Region....87
1.14.16 Class Topology....87
1.14.17 Class World Geodetic System <<Value>>....88
1.15 Concept Library::Objectives....89
1.15.1 Diagram: Objectives....89
1.15.2 Class Benefit89
1.15.3 Class Consequence....90
1.15.4 Association Class Consequence of Situation <<Relationship>>....90
1.15.5 Class Desirability Assessment....91
1.15.6 Class Means <<Role>>....92
1.15.7 Association Class Means To End <<Relationship>>....92
1.15.8 Class Objective....93
1.15.9 Association Class Objective of Stakeholder <<Relationship>>....93
1.15.10 Class Opportunity....94
1.15.11 Class Stakeholder <<Role>>....95
1.15.12 Association Class Stakeholder Desirability <<Relationship>>....95
1.16 Concept Library::Observations....97
1.16.1 Diagram: Measurement....97
1.16.2 Diagram: Observations....97
1.16.3 Association Context of Observation....98
1.16.4 Association Class Measurement <<Relationship>>....98
1.16.5 Association Class Observation <<Relationship>>....98
1.16.6 Class Observation Tool <<Role>>....99
1.16.7 Class Observer <<Role>>....99
1.17 Concept Library::Occurrences....100
1.17.1 Diagram: Occurrences....100
1.17.2 Class Activity....100
1.17.3 Class Actor 101
1.17.4 Class Actual Activity <<Intersection>>....101
1.17.5 Class Actual Occurrence <<Intersection>>....102
1.17.6 Class Occurrence....102
1.17.7 Association Class Output <<Relationship>>....102
1.17.8 Association Class Performance <<Relationship>>....103
1.17.9 Association Class Usage <<Relationship>>....103
1.18 Concept Library::Organizations....105
1.18.1 Diagram: Organization....105
1.18.2 Association Class Membership <<Relationship>>....105
1.18.3 Class Mission Objective....106
1.18.4 Class Organization....106
1.18.5 Class Organizational Unit <<Role>>....107
1.18.6 Class Parent Organization <<Role>>....107
1.18.7 Association Class Part of Organization <<Relationship>>....107
1.18.8 Class Program....108
1.19 Concept Library::Organizations::Corporations....109

1.19.1 Diagram: Corporations 109
1.19.2 Class Incorporated Organization 109
1.19.3 Association Class Incorporation <<Relationship>> 109
1.20 Concept Library::Organizations::Geopolitical Organizations 111
1.20.1 Diagram: Geopolitical Entities 111
1.20.2 Class Country....111
1.20.3 Class Country ID <<Value>> 111
1.20.4 Class Geopolitical Entity 112
1.20.5 Class Geopolitical ID <<Value>> 112
1.20.6 Class Geopolitical Region <<Role>> 112
1.20.7 Association Class Governing Authority <<Relationship>> 112
1.21 Concept Library::Parthood 113
1.21.1 Diagram: Parthood....113
1.21.2 Class Composite <<Role>> 113
1.21.3 Class Part <<Role>>....113
1.21.4 Association Class Parthood <<Relationship>>....113
1.22 Concept Library::Permissions 115
1.22.1 Diagram: Permission....115
1.22.2 Association Class Permission <<Relationship>> 115
1.23 Concept Library::Persons 117
1.23.1 Diagram: Person 117
1.23.2 Diagram: Person Identifiers 118
1.23.3 Diagram: Person Name Representations....119
1.23.4 Class Access Identifier <<Value>> 119
1.23.5 Class Financial Identifier <<Value>>....119
1.23.6 Class Managed Person Identifier <<Value>>....120
1.23.7 Class Passport Identifier <<Value>> 120
1.23.8 Class Person 120
1.23.9 Association Class Person at location <<Relationship>> 120
1.23.10 Class Person Name <<Value>> 121
1.23.11 Class Person Structured Name <<Value>> 121
1.23.12 Association Class Residency <<Relationship>>....122
1.23.13 Class Social Security Number <<Value>> 122
1.24 Concept Library::Physical Entities 123
1.24.1 Diagram: Physical Entities....123
1.24.2 Class Animal124
1.24.3 Class Conveyance 124
1.24.4 Class Device 125
1.24.5 Class Item 125
1.24.6 Class Managed Item Identifier <<Value>> 125
1.24.7 Class Physical Entity....125
1.24.8 Class Physical Feature 126
1.24.9 Class Physical Tool <<Role>> 126
1.24.10 Class Spatial Entity....126
1.24.11 Class Telecommunication Device....127
1.25 Concept Library::Places 129
1.25.1 Diagram: Place 129
1.25.2 Class Facility <<Role>> 129
1.25.3 Association Class Operating Location <<Relationship>> 129
1.25.4 Class Place <<Role>> 130
1.25.5 Association Class Place of Occurrance <<Relationship>>....130
1.25.6 Class Residence <<Role>>....131
1.26 Concept Library::Policies....132
1.26.1 Diagram: Policy 132
1.26.2 Association Class Assertion of Policy <<Relationship>> 132

1.26.3 Class Policy 133
1.27 Concept Library::Predictions 134
1.27.1 Diagram: Prediction....134
1.27.2 Association Class Prediction <<Relationship>>....134
1.27.3 Class Predictor <<Role>> 135
1.28 Concept Library::Processes....136
1.28.1 Diagram: Process....137
1.28.2 Class Composite Process <<Role>>....138
1.28.3 Association Class Invoke Process <<Relationship>>....138
1.28.4 Class Modus Operandi....138
1.28.5 Class Occurrence Pattern....138
1.28.6 Class Plan 139
1.28.7 Class Process Action....139
1.28.8 Association Class Process Decomposition <<Relationship>>....139
1.28.9 Association Class When <<Relationship>> 140
1.29 Concept Library::Processes::Capabilities....142
1.29.1 Diagram: Capabilities 142
1.29.2 Class Capability <<Role>>....143
1.29.3 Class Create 143
1.29.4 Class Damage 143
1.29.5 Class Destroy....143
1.29.6 Class Disrupt Process 143
1.29.7 Class Entry Action....144
1.29.8 Class Exit Action 144
1.29.9 Class Pause Process....144
1.29.10 Class Possible Actions....144
1.29.11 Class Stop Process....144
1.30 Concept Library::Processes::Composite Conditions 145
1.30.1 Diagram: Composite Condition 145
1.30.2 Class AND Condition....145
1.30.3 Class Composite Condition....145
1.30.4 Class OR Condition 146
1.30.5 Class XOR Condition 146
1.31 Concept Library::Processes::Scenarios 147
1.31.1 Diagram: Scenarios....147
1.31.2 Class Goal State....147
1.31.3 Class Scenario....148
1.31.4 Class Scenario Step....148
1.31.5 Association Class Step in Scenario <<Relationship>>....148
1.32 Concept Library::Quantities and Units....149
1.32.1 Diagram: Quantities and units....150
1.32.2 Class Confidence Metric <<Quantity Kind>> 150
1.32.3 Class Count <<Quantity Kind>> 151
1.32.4 Class Currency Benefit Metric <<Quantity Kind>> 151
1.32.5 Class Harm-Benefit Metric <<Quantity Kind>> 151
1.32.6 Class Metric <<Quantity Kind>> 151
1.32.7 Class Probability Metric <<Quantity Kind>>....152
1.32.8 Class Time Coordinate <<Quantity Kind>>....152
1.33 Concept Library::Quantities and Units::Quantity Kinds....156
1.33.1 Diagram: Quantity Kinds....156
1.33.2 Class Absorbed Dose (Radiation) <<Quantity Kind>> 156
1.33.3 Class Acceleration <<Quantity Kind>>....156
1.33.4 Class Amount of Substance <<Quantity Kind>>....157
1.33.5 Class Angle <<Quantity Kind>> 157
1.33.6 Class Area <<Quantity Kind>> 157

1.33.7 Class Color <<Quantity Kind>> 157
1.33.8 Class Concentration <<Quantity Kind>> 157
1.33.9 Class Concentration (amount of substance) <<Quantity Kind>> 157
1.33.10 Class Concentration (Mass) <<Quantity Kind>> 158
1.33.11 Class Concentration (Volume) <<Quantity Kind>> 158
1.33.12 Class Currency <<Quantity Kind>> 158
1.33.13 Class Dose Equivalent (Radiation) <<Quantity Kind>> 158
1.33.14 Class Duration <<Quantity Kind>> 158
1.33.15 Class Electric Current <<Quantity Kind>> 159
1.33.16 Class Electric Potential <<Quantity Kind>> 159
1.33.17 Class Energy <<Quantity Kind>> 159
1.33.18 Class Force <<Quantity Kind>> 159
1.33.19 Class Frequency <<Quantity Kind>> 159
1.33.20 Class Length <<Quantity Kind>> 160
1.33.21 Class Luminosity <<Quantity Kind>> 160
1.33.22 Class Mass <<Quantity Kind>> 160
1.33.23 Class Mass Density <<Quantity Kind>> 160
1.33.24 Class Physical Quantity <<Quantity Kind>> 160
1.33.25 Class Power <<Quantity Kind>> 161
1.33.26 Class Pressure <<Quantity Kind>> 161
1.33.27 Class Radiation Exposure <<Quantity Kind>> 161
1.33.28 Class Radioactivity <<Quantity Kind>> 161
1.33.29 Class Speed <<Quantity Kind>> 161
1.33.30 Class Temperature <<Quantity Kind>> 161
1.33.31 Class Volume <<Quantity Kind>> 162
1.34 Concept Library::Quantities and Units::Units 163
1.34.1 Diagram: Common Units 1 163
1.34.2 Diagram: Common Units 2 164
1.34.3 Class Acre <<Unit Value>> 164
1.34.4 Class Ampere <<Base Unit Value>> 164
1.34.5 Class Becquerel (Bq) <<Unit Value>> 164
1.34.6 Class Candela <<Base Unit Value>> 165
1.34.7 Class Celsius <<Unit Value>> 165
1.34.8 Class Compound Duration Value <<Unit Value>> 165
1.34.9 Class Concentration Percent <<Base Unit Value>> 165
1.34.10 Class Coulomb/kilogram (C/kg). <<Unit Value>> 165
1.34.11 Class Cubic Feet <<Unit Value>> 165
1.34.12 Class Cubic Inch <<Unit Value>> 165
1.34.13 Class Cubic Meter <<Base Unit Value>> 166
1.34.14 Class Cup (US) <<Unit Value>> 166
1.34.15 Class Curie (Ci) <<Base Unit Value>> 166
1.34.16 Class Day <<Unit Value>> 166
1.34.17 Class Degrees <<Unit Value>> 166
1.34.18 Class Fahrenheit <<Unit Value>> 166
1.34.19 Class Fluid Ounce (US) <<Unit Value>> 166
1.34.20 Class Foot <<Unit Value>> 167
1.34.21 Class Gallon (Imperial) <<Unit Value>> 167
1.34.22 Class Gallon (US) <<Unit Value>> 167
1.34.23 Class Gram <<Unit Value>> 167
1.34.24 Class Gray (Gy) <<Base Unit Value>> 167
1.34.25 Class Hertz <<Base Unit Value>> 167
1.34.26 Class Horsepower <<Unit Value>> 167
1.34.27 Class Hour <<Unit Value>> 168
1.34.28 Class Inch <<Unit Value>> 168
1.34.29 Class Joule <<Base Unit Value>> 168

1.34.30 Class Kelvin <<Base Unit Value>>....168
1.34.31 Class Kg per cubic meter <<Base Unit Value>>....168
1.34.32 Class Kilogram <<Base Unit Value>>....168
1.34.33 Class Kilogram per cubic meter <<Base Unit Value>>....168
1.34.34 Class Kilometer <<Unit Value>>....169
1.34.35 Class Kilometer per Hour <<Base Unit Value>>....169
1.34.36 Class Kilowatt hour <<Unit Value>>....169
1.34.37 Class Liquid Volume <<Quantity Kind>>....169
1.34.38 Class Meter <<Base Unit Value>>....169
1.34.39 Class Meter per second squared <<Base Unit Value>>....169
1.34.40 Class Mile <<Unit Value>>....170
1.34.41 Class Miles per Hour <<Unit Value>>....170
1.34.42 Class Millimeter <<Unit Value>>....170
1.34.43 Class Millisecond <<Unit Value>>....170
1.34.44 Class Minute <<Unit Value>>....170
1.34.45 Class Mole <<Base Unit Value>>....170
1.34.46 Class Mole Per Cubic Meter <<Base Unit Value>>....170
1.34.47 Class Month <<Unit Value>>....171
1.34.48 Class Newton <<Base Unit Value>>....171
1.34.49 Class Ounce-Mass (US) <<Unit Value>>....171
1.34.50 Class Pascal <<Base Unit Value>>....171
1.34.51 Class Pint (US) <<Unit Value>>....171
1.34.52 Class Pound-Force <<Unit Value>>....171
1.34.53 Class Pound-Mass (Imperial) <<Unit Value>>....171
1.34.54 Class Pound-Mass (US lb) <<Unit Value>>....172
1.34.55 Class PSI <<Unit Value>>....172
1.34.56 Class Quart (US) <<Unit Value>>....172
1.34.57 Class Radians <<Base Unit Value>>....172
1.34.58 Class Radiation Absorbed Dose (rad) <<Unit Value>>....172
1.34.59 Class Roentgen (R) <<Base Unit Value>>....172
1.34.60 Class Roentgen Equivalent Man (REM) <<Unit Value>>....172
1.34.61 Class Scalar Duration Value <<Unit Value>>....173
1.34.62 Class Second <<Base Unit Value>>....173
1.34.63 Class Sievert (Sv), <<Base Unit Value>>....174
1.34.64 Class Square Feet <<Unit Value>>....174
1.34.65 Class Square Meter <<Base Unit Value>>....174
1.34.66 Class Volt <<Base Unit Value>>....174
1.34.67 Class Watt <<Base Unit Value>>....174
1.34.68 Class Yard <<Unit Value>>....174
1.34.69 Class Year <<Unit Value>>....174
1.35 Concept Library::Resources....175
1.35.1 Diagram: Resource....175
1.35.2 Diagram: Resource Actions....176
1.35.3 Class Abuse Resource....176
1.35.4 Class Capture Resource....176
1.35.5 Class Damage Resource....176
1.35.6 Class Exceed Resource Capacity....176
1.35.7 Class Modify Resource....176
1.35.8 Class Performer <<Role>>....177
1.35.9 Class Resource <<Role>>....177
1.35.10 Class Resource Actions....177
1.35.11 Association Class Resource Dependency <<Relationship>>....177
1.35.12 Class Tool <<Role>>....178
1.36 Concept Library::Response....179
1.36.1 Diagram: Response....179

1.36.2 Association Class Initiation of Response <<Relationship>> 179
1.36.3 Class Response <<Role>> 179
1.37 Concept Library::Situations 181
1.37.1 Diagram: Actual Situations 181
1.37.2 Diagram: Situation Relationships 182
1.37.3 Diagram: Situation Timeframes 184
1.37.4 Diagram: Situations 185
1.37.5 Class Actual Situation <<Intersection>> 185
1.37.6 Class Actual State <<Intersection>> 186
1.37.7 Association Class Cause and Effect <<Relationship>> 186
1.37.8 Class Current Situation 187
1.37.9 Association Class Effect <<Relationship>> 187
1.37.10 Association Class Enablement <<Relationship>> 188
1.37.11 Association Class Involvement <<Relationship>> 189
1.37.12 Association Class Negation Effect <<Relationship>> 189
1.37.13 Class Past Situation 190
1.37.14 Class Potential Situation 190
1.37.15 Class Situation 191
1.37.16 Class Situation Type 191
1.37.17 Class State 191
1.37.18 Association State of Entity 191
1.38 Concept Library::Social Agents 193
1.38.1 Diagram: Social Agent 193
1.38.2 Diagram: Social Agent Identifiers 194
1.38.3 Association Class Assertion <<Relationship>> 194
1.38.4 Class Associate <<Role>> 194
1.38.5 Association Class Associated Actor <<Relationship>> 194
1.38.6 Class License Identifier <<Value>> 195
1.38.7 Class Local Identifier <<Value>> 195
1.38.8 Class Managed Social Agent Identifier <<Value>> 195
1.38.9 Class Member <<Role>> 196
1.38.10 Class Regional Identifier <<Value>> 196
1.38.11 Class Social Agent 196
1.38.12 Class Tax Authority Identifier <<Value>> 196
1.39 Concept Library::Systems 197
1.39.1 Diagram: System 197
1.39.2 Class Access Point 198
1.39.3 Class Boundary 198
1.39.4 Association Class Boundary of System <<Relationship>> 198
1.39.5 Association Class Opening in a Boundary <<Relationship>> 199
1.39.6 Association Class Point Of Entry <<Relationship>> 199
1.39.7 Class Subsystem <<Role>> 200
1.39.8 Class System 200
1.40 Concept Library::Time & Temporal Entities 201
1.40.1 Diagram: Time 202
1.40.2 Class Date and Time <<Quantity Kind>> 203
1.40.3 Class Date Coordinate <<Quantity Kind>> 203
1.40.4 Association Duration of Entity 203
1.40.5 Association Entity Exists for Interval 203
1.40.6 Association Finish Time 204
1.40.7 Association Overlaps in Time 205
1.40.8 Association Start Time 206
1.40.9 Class Temporal Composite <<Role>> 206
1.40.10 Class Temporal Entity <<Intersection>> 207
1.40.11 Association Temporal Order 207

1.40.12 Association Class Temporal Part <<Relationship>> 207
1.40.13 Class Temporal Region....208
1.40.14 Class Time Coordinate <<Quantity Kind>> 209
1.40.15 Class Time Interval....209
1.40.16 Class Time Point....210
1.40.17 Class Time Scale....210
1.40.18 Association Time Scale Granularity 210
1.40.19 Association Time Scale of Time Point 210
1.41 Concept Library::Time & Temporal Entities::ISO Time Scale 211
1.41.1 Diagram: ISO Time 211
1.41.2 Class Date Time Coordinate (ISO 8601) <<Unit Value>>....212
1.42 Concept Library::Time & Temporal Entities::XSD Time Scale 213
1.42.1 Diagram: XSD Time Scale 213
1.42.2 Class XSD Date <<Unit Value>>....213
1.42.3 Class XSD Date Time <<Unit Value>> 213
1.42.4 Class XSD Time <<Unit Value>>....214
1.43 Concept Library::Vendors and Producers 215
1.43.1 Diagram: Vendors and Producers....215
1.43.2 Class Client <<Role>> 215
1.43.3 Class Individual Product <<Role>> 215
1.43.4 Class Manufactured Thing <<Role>> 216
1.43.5 Class Producer <<Role>>....216
1.43.6 Class Product Kind 216
1.43.7 Association Class Product of Supplier <<Relationship>>....216
1.43.8 Association Class Production <<Relationship>> 217
1.43.9 Association Class Providing <<Relationship>> 217
1.43.10 Class Serial Number <<Value>>....218
1.43.11 Class Supplier <<Role>>....218
2 Threat-risk-conceptual-model....219
2.1.1 Diagram: Conceptual Threat/Risk Model Diagrams....219
2.2 Threat-risk-conceptual-model::.Summary Diagrams....220
2.2.1 Diagram: Countermeasures....220
2.2.2 Diagram: Entity Kinds 221
2.2.3 Diagram: indicator Fundamentals....221
2.2.4 Diagram: MEta Concepts....222
2.2.5 Diagram: Objectives 222
2.2.6 Diagram: Risk Fundamentals....223
2.2.7 Diagram: Risk Relationships 224
2.2.8 Diagram: Risk Treatment Fundamentals....225
2.2.9 Diagram: Threat Actor....225
2.2.10 Diagram: Vulnerability Essentials 226
2.2.11 Diagram: What is risk? 227
2.3 Threat-risk-conceptual-model::Threat and Risk Specific Concepts 228
2.3.1 Diagram: Threat and Risk Specific Concepts 229
2.4 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Accusations....230
2.4.1 Diagram: Accusations....230
2.4.2 Association Class Accusation <<Relationship>> 230
2.5 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Campaigns 231
2.5.1 Diagram: Campaign....231
2.5.2 Diagram: Campaign Objectives....232
2.5.3 Class Campaign 232
2.6 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Danger 233
2.6.1 Diagram: Danger 234
2.6.2 Class Attack <<Role>>....235
2.6.3 Class Indirect Threat <<Role>> 235

2.6.4 Class Natural Threat <<Role>> 235
2.6.5 Association Class Target of Attack <<Relationship>> 235
2.6.6 Class Threat <<Role>> 236
2.6.7 Class Unintentional Threat <<Role>> 236
2.7 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Danger Sources 237
2.7.1 Diagram: Danger Sources 237
2.7.2 Class Accident <<Role>> 237
2.7.3 Association Class Contribution to Danger <<Relationship>> 238
2.7.4 Class Danger Source <<Role>> 238
2.7.5 Class Dangerous Condition <<Role>> 238
2.7.6 Class Dangerous Event <<Role>> 239
2.7.7 Association Class Exploit of Vulnerability <<Relationship>> 239
2.7.8 Class Objective to Disrupt 239
2.7.9 Class Unwitting Participant <<Role>> 239
2.8 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Danger Sources::Source of Danger Categories 241
2.8.1 Diagram: Source of Danger Categories 241
2.8.2 Class Biological Danger <<Category>> 241
2.8.3 Class CBRN Danger <<Category>> 242
2.8.4 Class Chemical Danger <<Category>> 242
2.8.5 Class Civil Unrest Danger <<Category>> 242
2.8.6 Class Criminal Danger <<Category>> 242
2.8.7 Class Cyber Danger <<Category>> 242
2.8.8 Class Fire Danger <<Category>> 242
2.8.9 Class Geophysical Danger <<Category>> 242
2.8.10 Class Loss of Control Danger <<Category>> 242
2.8.11 Class Meteorological Danger <<Category>> 243
2.8.12 Class Nuclear Danger <<Category>> 243
2.8.13 Class Radiological Danger <<Category>> 243
2.8.14 Class Safety Danger <<Category>> 243
2.8.15 Class Security Danger <<Category>> 243
2.8.16 Class Source of Danger Category <<Category>> 243
2.8.17 Class Terrorism Danger <<Category>> 243
2.8.18 Class War Danger <<Category>> 243
2.9 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents 244
2.9.1 Diagram: Incident 244
2.9.2 Association Class Danger Leads to Incident <<Relationship>> 244
2.9.3 Class Incident <<Role>> 245
2.9.4 Class Witness <<Role>> 245
2.9.5 Association Class Witnessing <<Relationship>> 245
2.10 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Failure 247
2.10.1 Diagram: Failure 247
2.10.2 Class Failure 247
2.10.3 Association Class Failure of Resource <<Relationship>> 247
2.11 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Failure Categories249
2.11.1 Diagram: Failure Categories 249
2.11.2 Class Access Control Failure <<Category>> 249
2.11.3 Class Control Failure <<Category>> 249
2.11.4 Class Cyber System Failure <<Category>> 250
2.11.5 Class Failure Category <<Category>> 250
2.11.6 Class Industrial Control Failure <<Category>> 250
2.11.7 Class Physical System Failure <<Category>> 250
2.11.8 Class Process Failure <<Category>> 250
2.11.9 Class System Failure <<Category>> 250
2.12 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Impact Categories251

2.12.1 Diagram: Impact Categories 251
2.12.2 Class Communications Impact <<Category>> 251
2.12.3 Class Compliance Impact <<Category>> 252
2.12.4 Class Control of Information Impact <<Category>> 252
2.12.5 Class Decision-making Impact <<Category>> 252
2.12.6 Class Disinformation Impact <<Category>> 252
2.12.7 Class Electromagnetic Spectrum Impact <<Category>> 252
2.12.8 Class Environmental Impact <<Category>> 252
2.12.9 Class Financial Impact <<Category>> 252
2.12.10 Class Health Impact <<Category>> 252
2.12.11 Class Identity Spoof Impact <<Category>> 253
2.12.12 Class Image Impact <<Category>> 253
2.12.13 Class Impact Category <<Category>> 253
2.12.14 Class Information Availability Impact <<Category>> 253
2.12.15 Class Information Impact <<Category>> 253
2.12.16 Class Infrastructure Impact <<Category>> 253
2.12.17 Class Intellectual Property Impact <<Category>> 253
2.12.18 Class Legal Impact <<Category>> 253
2.12.19 Class Mission Impact <<Category>> 254
2.12.20 Class Non-Technical Impact <<Category>> 254
2.12.21 Class Safety Impact <<Category>> 254
2.12.22 Class Transport Impact <<Category>> 254
2.13 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Incident Response255
2.13.1 Diagram: Incident Response 255
2.13.2 Class Incident Containment <<Role>> 255
2.13.3 Class Incident Recovery <<Role>> 255
2.13.4 Class Incident Response <<Role>> 255
2.14 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Indicators 257
2.14.1 Diagram: Indicator 257
2.14.2 Diagram: Observability 258
2.14.3 Diagram: Sighting 258
2.14.4 Class Blacklist Indicator 258
2.14.5 Class Indicator 259
2.14.6 Association Class Indicator Indicates Situation <<Relationship>> 259
2.14.7 Class Indicator Pattern 260
2.14.8 Class Indicator Watchlist 260
2.14.9 Association Class Observability <<Relationship>> 260
2.14.10 Association Class Scope of Indicator <<Relationship>> 260
2.14.11 Association Class Sighting <<Relationship>> 261
2.14.12 Association Class Sighting Indicates Situation <<Relationship>> 261
2.14.13 Association Class Sighting Matches Indicator <<Relationship>> 262
2.14.14 Association Class Watch <<Relationship>> 263
2.14.15 Class Whitelist Indicator 264
2.15 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Risk 265
2.15.1 Diagram: Risk 265
2.15.2 Diagram: Risk Metrics 266
2.15.3 Diagram: Risk Objectives 267
2.15.4 Association Class Impose Strategy <<Relationship>> 267
2.15.5 Class Objective forSafety and Security 268
2.15.6 Class Objective to Protect Assets 268
2.15.7 Class Risk 268
2.15.8 Class Risk Mitigation Strategy 269
2.15.9 Class Risk Owner <<Role>> 269
2.15.10 Class Risk Reduction Objective 269
2.15.11 Association Class Risk To Resource <<Relationship>> 269

2.15.12 Association Class Risk Topic <<Relationship>> 270
2.15.13 Association Class Stakeholder Risk <<Relationship>> 271
2.15.14 Class Threat Likelihood <<Quantity Kind>> 272
2.15.15 Association Class Valuation of Asset <<Relationship>> 272
2.15.16 Class Valued Asset <<Role>> 272
2.16 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Risk Treatments 274
2.16.1 Diagram: Risk Treatment 274
2.16.2 Diagram: Safeguard Monitoring 275
2.16.3 Class Accept Risk 275
2.16.4 Association Class Assume Risk <<Relationship>> 275
2.16.5 Class Avoid Danger 276
2.16.6 Class Countermeasure <<Role>> 276
2.16.7 Association Class Countermeasure for Strategy <<Relationship>> 276
2.16.8 Association Class Countermeasure Mitigates <<Relationship>> 277
2.16.9 Class Defensive Step 278
2.16.10 Class Management Control 278
2.16.11 Class Mitigation Actor <<Role>> 278
2.16.12 Association Class Monitor <<Relationship>> 278
2.16.13 Class Monitoring Control 279
2.16.14 Class Operational Control 279
2.16.15 Association Class Protection <<Relationship>> 279
2.16.16 Class Risk Agent <<Role>> 280
2.16.17 Association Class Risk Treatment <<Relationship>> 280
2.16.18 Class Risk Treatment Strategy 280
2.16.19 Association Class Safeguarding <<Relationship>> 281
2.16.20 Class Technical Control 281
2.16.21 Class Transfer Risk 281
2.17 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Threat Actors 282
2.17.1 Diagram: Threat Actors 282
2.17.2 Class Disrupt Stakeholder's Objective 282
2.17.3 Class Disruptive Action <<Role>> 283
2.17.4 Class Disruptive Step 283
2.17.5 Association Class Perpetrate <<Relationship>> 283
2.17.6 Class Threat Actor <<Role>> 284
2.18 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Undesirable Situations 285
2.18.1 Diagram: Undesirable Situations 286
2.18.2 Class Harm 286
2.18.3 Association Class Harms Resource <<Relationship>> 287
2.18.4 Association Class Harms Victim <<Relationship>> 288
2.18.5 Association Class Source of Harm <<Relationship>> 289
2.18.6 Class Undesirable Condition <<Role>> 289
2.18.7 Class Undesirable Event <<Role>> 289
2.18.8 Class Undesirable Situation <<Role>> 289
2.18.9 Class Victim <<Role>> 290
2.19 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities 291
2.19.1 Diagram: Vulnerability 291
2.19.2 Diagram: Vulnerability Identifiers 292
2.19.3 Class Physical Vulnerability 292
2.19.4 Association Class Supporting Condition <<Relationship>> 292
2.19.5 Class Vulnerability 293
2.19.6 Class Vulnerability Identifier <<Value>> 293
2.19.7 Class Vulnerability Metric <<Quantity Kind>> 293
2.19.8 Association Class Vulnerability of Resource <<Relationship>> 294
2.20 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Cyber Vulnerabilities 295

2.20.1 Diagram: Cyber Vulnerability 295
2.20.2 Class Communications Vulnerability....295
2.20.3 Class CVE Identifier <<Value>> 296
2.20.4 Class Cyber Vulnerability....296
2.20.5 Class Information System Vulnerability 296
2.20.6 Class Information Vulnerability....296
2.20.7 Class OSVDB Identifier <<Value>> 296
2.20.8 Class Software Vulnerability 296
2.21 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors 297
2.21.1 Diagram: Vulnerability Vectors....298
2.21.2 Class CVSS Score <<Quantity Kind>> 299
2.22 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Weapons 307
2.22.1 Diagram: Weapons 307
2.22.2 Class Cyber Weapon <<Role>> 307
2.22.3 Class Physical Weapon <<Role>> 307
2.22.4 Class Weapon <<Role>>....307
2.22.5 Association Class Weapon Leverages Vulnerability <<Relationship>> 308
3 SMIF Conceptual Model 308
3.1 Diagram: SMIF Packages....309
3.2 SMIF Conceptual Model::Associations....311
3.2.1 Diagram: Associations....311
3.2.2 Class Association....312
3.2.3 Class Association Type....312
3.2.4 Class Binary Association 312
3.2.5 Class Binary Association Type 312
3.2.6 Class Restriction Type 313
3.3 SMIF Conceptual Model::Constraints....314
3.3.1 Diagram: General Rules....314
3.3.2 Diagram: Property Constraints 315
3.3.3 Diagram: Rules in Context....316
3.3.4 Diagram: Rules Summary....317
3.3.5 Diagram: Type Constraints 318
3.3.6 Class Conditional....319
3.3.7 Class Conditional Rule <<Rule>> 319
3.3.8 Class Constraint <<Rule>>....319
3.3.9 Association Constraint Covers....319
3.3.10 Class Covering Constraint <<Rule>> 319
3.3.11 Class Disjoint <<Rule>> 319
3.3.12 Class Enumerated <<Rule>> 320
3.3.13 Class Equivalent <<Rule>> 320
3.3.14 Class Multiplicity Constraint <<Rule>>....320
3.3.15 Association Multiplicity Target 321
3.3.16 Class Property Constraint <<Rule>> 321
3.3.17 Class Property Transitivity Constraint <<Rule>>....322
3.3.18 Association Proposition Subsumption 322
3.3.19 Association Rule Constrains 322
3.3.20 Class Type Constraint <<Rule>> 322
3.3.21 Association Unique Set....323
3.3.22 Class Uniqueness Constraint <<Rule>> 323
3.4 SMIF Conceptual Model::Expressions....324
3.4.1 Diagram: Expressions....324
3.4.2 Class And Function....324
3.4.3 Class Binary Boolean Function 324
3.4.4 Class Binary Function....325

3.4.5 Class Boolean Function 325
3.4.6 Class Constant Reference 325
3.4.7 Association Constant Value 325
3.4.8 Class Current Context 325
3.4.9 Class Equality Function 326
3.4.10 Class Expression Context 326
3.4.11 Association Expression Evaluation 326
3.4.12 Class Expression Node 326
3.4.13 Class Function Call 327
3.4.14 Association Function Called 327
3.4.15 Class Function Type 327
3.4.16 Class Not Function 327
3.4.17 Class Object Operation Type 327
3.4.18 Association OO Target 328
3.4.19 Class Or Function 328
3.4.20 Class Traversal 328
3.4.21 Association Traverse Through 328
3.4.22 Class Xor Function 329
3.5 SMIF Conceptual Model::Facets 330
3.5.1 Diagram: Facets 330
3.5.2 Class Category Type 330
3.5.3 Association Facet Constraint 330
3.5.4 Association Facet Of Entity 331
3.5.5 Class Facet Type 331
3.5.6 Class Phase Type 332
3.5.7 Class Role Type 332
3.6 SMIF Conceptual Model::Identifiers 333
3.6.1 Diagram: Identifier Summary 333
3.6.2 Diagram: Identifiers 334
3.6.3 Association Identification 334
3.6.4 Class Identifier <<Value>> 335
3.6.5 Association Identifier in Namespace 335
3.6.6 Class IRI Identifier <<Value>> 335
3.6.7 Class Name <<Value>> 336
3.6.8 Class Namespace 336
3.6.9 Association Naming 336
3.6.10 Association Prefered Identification 337
3.6.11 Class Technical Identifier <<Value>> 337
3.6.12 Class Term <<Value>><<Intersection>> 337
3.6.13 Class Text Identifier <<Value>> 337
3.6.14 Class Unique Identifier <<Value>> 338
3.6.15 Class Unique Text Identifier <<Value>><<Intersection>> 338
3.7 SMIF Conceptual Model::Lexical Scope 339
3.7.1 Diagram: Lexical Scope 339
3.7.2 Class Conceptual Package 339
3.7.3 Association Definition 340
3.7.4 Class Include340
3.7.5 Class Information Repository 340
3.7.6 Class Lexical Context 340
3.7.7 Class Lexical Reference 340
3.7.8 Class Lexical Scope 341
3.7.9 Class Logical Package 341
3.7.10 Class Model 341
3.7.11 Class Package 341
3.7.12 Class Physical Package 341

3.7.13 Class Prefix <<Value>> 341
3.7.14 Association Prefix Text 342
3.7.15 Association Repository Conteining Model 342
3.7.16 Association Scope of Reference 342
3.7.17 Association Scope Reference....342
3.7.18 Association Statement....342
3.8 SMIF Conceptual Model::Mapping 344
3.8.1 Diagram: Facades 344
3.8.2 Diagram: Mapping Rules....345
3.8.3 Class Computed Facade....345
3.8.4 Association Concrete Map End 346
3.8.5 Association Concrete Pattern Body 346
3.8.6 Class Facade 346
3.8.7 Association Map Rule Type Assertion 346
3.8.8 Association Mapped variable....346
3.8.9 Class Mapping <<Rule>>....347
3.8.10 Class Match End....347
3.8.11 Class Match Rule <<Rule>>....347
3.8.12 Association Match Rules 348
3.8.13 Association Reference Map End....348
3.8.14 Association Reference Pattern Body....348
3.8.15 Association Representation....348
3.8.16 Class Representation Rule <<Rule>> 349
3.8.17 Association Represented Concept....349
3.9 SMIF Conceptual Model::Metadata 350
3.9.1 Diagram: Metadata 350
3.9.2 Class Definition <<Value>>....350
3.9.3 Association Definition Relationship 351
3.9.4 Class Information Source <<Role>> 351
3.9.5 Class Metadata <<Value>> 351
3.9.6 Association Metadata relationship....352
3.9.7 Association Source of Information 352
3.9.8 Class Speech Act <<Value>>....352
3.10 SMIF Conceptual Model::Patterns....353
3.10.1 Diagram: Patterns 353
3.10.2 Class Computed....353
3.10.3 Association Exclusion....354
3.10.4 Class Expression Variable 354
3.10.5 Class Focus Variable 354
3.10.6 Class Part Variable 354
3.10.7 Class Pattern <<Intersection>>....354
3.10.8 Association Pattern Bindings....355
3.10.9 Class Pattern Match 355
3.10.10 Association Pattern Matches....355
3.10.11 Class Pattern of Type....355
3.10.12 Class Pattern Variable....355
3.10.13 Association Pattern Variables....356
3.10.14 Class Proposition Variable 356
3.10.15 Association Qualified Proposition 356
3.10.16 Association Subject of Pattern Relationship 357
3.10.17 Association Subsetting....357
3.10.18 Class Type Pattern Variable 357
3.10.19 Class Variable Binding 357
3.11 SMIF Conceptual Model::Properties....360
3.11.1 Diagram: Characteristics....360

3.11.2 Diagram: Properties 361
3.11.3 Diagram: Property Foundation 362
3.11.4 Class Abstract Property Owner 362
3.11.5 Class Annotation Property 362
3.11.6 Association Bound Individual 362
3.11.7 Association Bound Property 363
3.11.8 Association Bound Subject 363
3.11.9 Class Characteristic Binding 363
3.11.10 Class Characteristic Type 364
3.11.11 Class Computed Property Type 364
3.11.12 Class Concrete Property Type 364
3.11.13 Class Derived Property Type 364
3.11.14 Class Identifiable Property Owner 365
3.11.15 Class Owned Property Binding 365
3.11.16 Class Owned Property Type 365
3.11.17 Class Property Binding 365
3.11.18 Association Property Domain 366
3.11.19 Class Property Owner Type 366
3.11.20 Association Property Range 366
3.11.21 Class Property Restriction 367
3.11.22 Class Property Type 367
3.11.23 Association Property Type Ownership 368
3.11.24 Association Proxy Delegate 368
3.11.25 Class Proxy Property Type 368
3.11.26 Association Redefinition 368
3.12 SMIF Conceptual Model::Relationships 369
3.12.1 Diagram: Relationships 369
3.12.2 Class Relationship 369
3.12.3 Class Relationship Type 369
3.13 SMIF Conceptual Model::Top level 371
3.13.1 Diagram: Context 371
3.13.2 Diagram: Top Level 372
3.13.3 Class Actual Entity 372
3.13.4 Association Assertion 373
3.13.5 Class Context 374
3.13.6 Class Entity 375
3.13.7 Association Extent of Context 376
3.13.8 Class Fact 376
3.13.9 Association Negation 376
3.13.10 Class Proposition 376
3.13.11 Class Scoped Context 377
3.13.12 Class Thing <<Anything>> 377
3.14 SMIF Conceptual Model::Types 379
3.14.1 Diagram: Type-instance 379
3.14.2 Diagram: Types 380
3.14.3 Diagram: Types as Context 381
3.14.4 Class Entity Type 381
3.14.5 Association Extent of Type 382
3.14.6 Association Generalization 382
3.14.7 Class Intersection Type 383
3.14.8 Association Sufficient Generalization 383
3.14.9 Class Type 383
3.14.10 Class Union Type 384
3.15 SMIF Conceptual Model::Values 385
3.15.1 Diagram: Values 386

3.15.2 Diagram: Values Only 387
3.15.3 Class Base Unit Type.....387
3.15.4 Class Quantity kind.....387
3.15.5 Association Referenced System of Units 388
3.15.6 Class Scalar Quantity <<Value>> 388
3.15.7 Class Structured Value <<Value>> 388
3.15.8 Class Structured Value Type 388
3.15.9 Class System of Units 388
3.15.10 Class Unit Type 389
3.15.11 Class Unit Value <<Value>>.....389
3.15.12 Class Value 390
3.15.13 Class Value Type.....390

1 Concept Library

Concepts that are common across many domains and purposes such that they may be used as needed to federate and translate information into a threat, risk or other domain. Also known as "Micro theories" in logic.

1.1.1 Diagram: Federated Concept Library

Figure 1. Federated Concept Library

Modules (as UML packages) defined as generic concepts under threat & risk. Each module focuses on a particular concept with dependencies on other modules. This forms a graph of "micro theories" that can be independently utilized.

1.1.2 Diagram: Foundation

Figure 2. Foundation

1.2 Concept Library::Abilities

The Ability module defines the basic concept of an *Ability* as the availability of a resource to an actor. The resource may be specific, such as $5000, a weapon, or general, such as the ability to teach math. Each Ability is a state indicating that it has a lifetime and can participate in all the state/situation and entity relations. *Credentials* may be physical or virtual and attest to the Ability.

Abilities may be created, enhanced, diminished, or eliminated with an *Alter Ability* Event. When an actor alters an Ability (of themselves or others) they are playing the role of a *Facilitator* who performs an *Alter Capacity*.

1.2.1 Diagram: Ability

Figure 3. Ability

1.2.2 Diagram: Alter ability or control

Figure 4. Alter ability or control

1.2.3 Association Class Ability <<Relationship>>

An Ability is the availability of a resource to an actor to perform activities.
Note that as with any entity, an Ability can also be categorized as a resource and thus the subject of other relationships..
Note that resources can be physical (such as a hammer), activities, virtual or more abstract such as training or experience.
[ISO/IEC 17027:2014] ability: capacity to perform an activity
[NIEM] CapabilityType:

Figure 1. Ability Detail

Direct Supertypes

Impact, Resource

Association Ends

can be utilized by : Actor [*]

The actor having the ability to utilize a resource for a purpose.

has ability to utilize : Capability [*]

A resource an actor can employ as part of a capability.

1.2.4 Association Class Affected Available Resource <<Relationship>>

An alter ability action affects an actors control over one or more resources. This relationship defines the set of resources impacted by an alter ability action.

Figure 2. Affected Available Resource

Direct Supertypes

Effect

Association Ends

affects control of : Resource [1..*]

Resource that an alter ability action impacts in terms of its availably to actors.

affected by action : Alter Capability [*]

Actions which impact the availability of a resource to actors.

1.2.5 Class Alter Capability

The Event of providing or removing abilities (including control) by impacting the resources available to an actor.

Direct Supertypes

Capability

1.2.6 Class Facilitator <<Role>>

An actor able to provide an ability to another actor. e.g., Joe has supervisor rights to a database.

Direct Supertypes

Actor

1.2.7 Class Lose Ability

An Event that reduces of the ability of an actor.

Direct Supertypes

Alter Capability

1.2.8 Class Obtain Ability

An Event that increases the resources available to an actor.

Direct Supertypes

Alter Capability

1.2.9 Association Class Strenthened Actor <<Relationship>>

The act of obtaining control of an entity.

Figure 3. Strenthened Actor

Direct Supertypes

Effect

Association Ends

obtains control : Actor [1]

Control obtained by a obtain control action.

obtained by : Obtain Ability [0..*]

Method by which control is obtained.

1.2.10 Class Transfer Ability

The purposeful or accidental transfer of ability or control from one actor to another. Such transfer of ability or control may be by agreement or force.

Direct Supertypes

Lose Ability, Obtain Ability, Occurrence

1.2.11 Association Class Weakened Actor <<Relationship>>

Relationship describing how an actor reduces the resources available to an actor.

Figure 4. Weakened Actor

Direct Supertypes

Effect

Association Ends

looses control : Actor [1]

Control that is lost as a result of a lose control Event.

lost by : Lose Ability [0..*]

Action that causes a loss of control.

1.3 Concept Library::Assertions

1.3.1 Diagram: Assertions

Figure 5. Assertions

1.4 Concept Library::Assessments

Concepts relating to the structured evaluation of an entity or entities.

1.4.1 Diagram: Assessment

Figure 6. Assessment

The evaluation or estimation of the nature, quality, or ability of someone or something.
[BMM] Assessment: judgment that an influencer affects the employment of means and/or the achievement of ends

1.4.2 Class Assessed Entity <<Role>>

Role of an entity that is assessed.

Direct Supertypes

Entity

1.4.3 Class Assessment

An evaluation, appraisal, or assessment of something or someone.

[NIEM] AssessmentType: The act of evaluating or estimating the nature, ability, or quality of something. {Note: In NIEM the assessment and activity are combined}

Direct Supertypes

Actual Entity

Attributes

assessment score : Metric

An evaluation score of an assessment.[NIEM]

1.4.4 Class Assessment Activity

An activity that results in an assessment - An evaluation, appraisal, or assessment of something or someone. An assessment frequently as an artifact, a report of the assessment.

[NIEM] AssessmentType: The act of evaluating or estimating the nature, ability, or quality of something.

Direct Supertypes

Activity

1.4.5 Association Class Entity Assessment <<Relationship>>

Entities assessed is a relationship between an assessment activity and an entity being assessed that is the topic of an assessment report.

Figure 7. Entities Assessed

Direct Supertypes

Temporally Related

Association Ends

assesses : Assessed Entity [1..*]

Entity assessed by an assessment activity

assessed by : Assessment [0..*]

Entity performing an assessment.

1.5 Concept Library::Contact Information

The definition of various ways to contact an entity. Subtypes of contact information supply specific formats.

1.5.1 Diagram: Contact Information

Figure 8. Contact Information

1.5.2 Class Communications Security Level <<Value>>

An abstract type for levels of security in communications.

Direct Supertypes

Text Identifier

1.5.3 Class Contact Means <<Value>>

Anything that may be used to communicate with an individual.
[FIBO] AddressingScheme
[NIEM] ContactMeans & ContactInformationType

Direct Supertypes

Unique Identifier

Attributes

security level : Communications Security Level
The level of security asserted as provided by the subject contact means. May default to the security level of the communications network.

1.5.4 Association Class Contact Via <<Relationship>>

Information relative to communicating with an entity.
[NIEM] ContactInformationAssociationType

Figure 9. Contact Via

Direct Supertypes

Relationship

Association Ends

contact via : Contact Means [*]

A way to contact an actor or place.
[FIBO] hasAddress (More specific concept - restricted to Postal Address)

contact for : Contactable [1..*]

An actor or place for which the contact information may be used to contact that entity.

Attributes

purpose : Contact Purpose
Purposes for contacting an entity, primarily work and personal.
[NIEM] ContactPurpose

availability : Contact Availability
An enumeration of the times contact information may be used.
[NIEM] ContactInformationAvailability

1.5.5 Class Contactable

Anything that can be send or receive information or be the proxy for things that can send or receive information, e.g., people, organizations and places.

Direct Supertypes

Actual Entity

1.5.6 Class Electronic Contact <<Value>>

Contact information that enables communications with or via an actor or telecommunications device by electronic means.
[FIBO] VirtualAddress: an address identifying a virtual, i.e. non-physical location

Direct Supertypes

Contact Means

Attributes

mobile : Boolean
Indicator that a contact method is mobile - not fixed to a location.

voice capable : Boolean
An indication that a contact method is voice capable.

document capable : Boolean
An indication that a contact method is capable of receiving documents, e.g., email.

text message capable : Boolean
An indication that a contact method is capable of receiving text messages of limited length.

fax capable : Boolean
Contact method for a communications device that is fax capable.

video capable : Boolean
An indication that a contact method is video capable.

1.5.7 Class Email Address <<Value>>

Contact information for the delivery of mail via an electronic network.
[NIEM] ContactEmailId (of <electronic contact>)

Direct Supertypes

Internet Contact

1.5.8 Class Internet Contact <<Value>>

[NIEM] A means of contact that provides for the digital electronic transmission of information via the Internet or a private network.

Direct Supertypes

Electronic Contact

Attributes

electronic address : Network Identifier
Electronic address by which to contact an entity via the Internet.

1.5.9 Class Network Identifier <<Value>>

A value used to identify a node in an electronic network.
[NIEM] ElectronicAddressType

Direct Supertypes

Technical Identifier

1.5.10 Class Postal Address <<Value>>

An address able to be used to deliver physical mail which may or may not represent a static physical location.
[FIBO] PPostalAddress: a physical address where postal communications can be addressed, for any kind of organization or person.
[NIEM] AddressType

Direct Supertypes

Contact Means, Location Identifier

1.5.11 Class Postal Address Structured <<Value>>

A structured representation of a postal address.

Direct Supertypes

Postal Address

Attributes

recipient name : Name =
Name of the recipient in a postal address which defaults to the name of the entity having the address. Should default to the contact for "has name".
[NIEM] AddressRecipientName

country ID : Country ID [0..1]
Postal country identifier.

post code : Postal Code [0..1]

[OGC] An address component which represents the identification of a subdivision of addresses and postal delivery points in a country, region, or city for postal purposes.
[NIEM] LocationPostalCode

state_province ID : Geopolitical ID [0..1]
Postal state identifier for a geopolitical regions.
[NIEM] LocationState

county ID : Geopolitical ID [0..1]
Postal county identifier.
[NIEM] LocationCounty

region ID : Location ID [*]
Postal region identifier.
[NIEM] AddressUrbanizationName

city ID : Geopolitical ID [0..1]
Postal city identifier.
[NIEM] LocationCityName

street ID : Location ID [0..1]
Postal street identifier.
[NIEM] AddressDeliveryPoint

place ID : Location ID [0..1]
Postal identifier for a specific place: House, building, facility, etc.

unit ID : Location ID [0..1]
Postal province identifier.
[NIEM] AddressSecondaryUnitText

post box ID : Postal Code [0..1]
A code defined for the purposes of delivering physical mail to a specific addresses.
[NIEM] AddressPrivateMailboxText

1.5.12 Class Postal Address Text <<Value>>

A textual representation of a postal address.

Direct Supertypes

Postal Address

Attributes

postal address : String Value [1]
Textual postal address for the delivery of mail.
[NIEM] AddressFullText

1.5.13 Class Postal Code <<Value>>

A code defined for the purposes of delivering physical mail to a set of addresses. "Zip code" in the U.S.
[OGC] An address component which represents the identification of a subdivision of addresses and postal delivery points in a country, region or city for postal purposes.
[FIBO] PostalCodeArea

Direct Supertypes

Text Identifier

1.5.14 Class Private Network Contact <<Value>>

Contact identifiers valid within a private network.

Direct Supertypes

Internet Contact

1.5.15 Class Radio Contact <<Value>>

Identifier for contact via radio.
[NIEM] ContactRadioType

Direct Supertypes

Electronic Contact

Attributes

channel : Network Identifier
Radio channel used for communications.
[NIEM] ContactRadioChannelText

call sign : Technical Identifier
Radio or user call sign used for radio communications.
[NIEM] ContactRadioCallSignID

1.5.16 Class Social Network Contact <<Value>>

Contact information to be used via a social network.
[NIEM] InstantMessageType
--InstanceMessengerServiceName = <has name>
--InstanceMessengerScreenId = "electronic address"

Direct Supertypes

Internet Contact

1.5.17 Class Telephone Area Code <<Value>>

A three-digit number that identifies one of the telephone service regions into which the US, Canada, and certain other countries are divided and that is dialed when calling from one area to another.

Direct Supertypes

Text Identifier

Attributes

value : Integer Value [0..999]
3 digit area code.

1.5.18 Class Telephone Country Code <<Value>>

2 digit Telephone codes for contacting people and organizations within countries.

Direct Supertypes

Country ID, Text Identifier

Attributes

value : Integer Value [0..99]

Country code digits.

1.5.19 Class Telephone Number <<Value>>

A way to contact an actor via a telephone.
[NIEM] TelephoneNumberType

Direct Supertypes

Electronic Contact

1.5.20 Class Telephone Number Structured <<Value>>

Structured representation of a telephone number.
[NIEM] NANPTelephoneNumberType & InternationalTelephoneNumberType

Direct Supertypes

Telephone Number

Attributes

◇ country code : Telephone Country Code [0..1]
Telephone country code.
[NIEM] TelephoneCountryCodeID

◇ area code : Telephone Area Code [0..1]
Telephone area code.
[NIEM] TelephoneAreaCodeID

◇ telephone exchange : Integer Value [0..1]
Number identifying a telephone exchange.
[NIEM] TelephoneExchangeID

◇ line [1]
Telephone line number.
[NIEM] TelephoneLineID

◇ telephone extension [0..1]
Telephone extension number.

1.5.21 Class Telephone Number Text <<Value>>

Unstructured (text) representation of a telephone number.
[NIEM] FullTelephoneNumberType

Direct Supertypes

Telephone Number

Attributes

◇ telephone number : String [1]
Textual telephone number.
[NIEM] TelephoneNumberFullID

1.5.22 Class Website Contact <<Value>>

A website that can be used to contact an individual.

Direct Supertypes

Internet Contact

Attributes

- electronic address : IRI Identifier

Electronic address by which to contact an entity via a website.
[NIEM] ContactWebsiteURI

1.5.221 Enumeration Contact Availability

A data type for a period of time or a situation in which an entity is available to be contacted with the given contact information.[NIEM]
package Concept Library::Contact Information
public enum Contact Availability
{night, day, emergency, evening, primary, secondary}

Literals

- night

Late night contact.

- day

Daytime contact.

- emergency

Emergency contact.

- evening

Late day or early night contact.

- primary

Primary contact.

- secondary

Secondary or alternate contact.

1.5.222 Enumeration Contact Purpose

Possible purposes for contact information.[NIEM]
package Concept Library::Contact Information
public enum Contact Purpose
{personal, work, other}

Literals

- personal

Personal communications.

- work

Work communications.

- other

Communications other than work or personal.

1.6 Concept Library::Containment

1.6.1 Diagram: Containment

Figure 10. Containment

1.6.2 Diagram: Move Between Containers

Figure 11. Move Between Containers

1.6.3 Diagram: Physical Containment

Figure 12. Physical Containment

1.6.4 Class Add To Container Event

An event that puts things into containers or locations. This results in a new containment relationship.

Direct Supertypes

Containment Event

1.6.5 Class Container <<Role>>

Role of something as a container for other things. Containers & Containment may be physical or virtual.

Direct Supertypes

Actual Entity

1.6.6 Association Class Containment <<Relationship>>

Relationship between a container and a contained thing.
Containment may or may not correspond to a part-of relationship. In some cases both the container (a jar) and what it contains (jelly beans) may be considered parts of a whole (a jar of jelly beans).

Figure 13. Containment Detail

Direct Supertypes

Temporally Related

Association Ends

contains : Actual Entity [*]

A thing contained within a container.

is in : Container [1..*]

A container which holds one or more contained things.

[FIBO] isLocatedAt: a property linking something to a location or place, which might be physical or virtual

1.6.7 Class Containment Event

An event impacting containment of the <contained thing>. Subtypes include "Put Into Event", "Removal Event" and "Relocation".

Direct Supertypes

Capability

1.6.8 Class Physical Container <<Role>>

A physical thing or location that contains other physical things or locations.

[DOLCE] Spacial Location

Direct Supertypes

Container, Spatial Entity

1.6.9 Association Class Physical Containment <<Relationship>>

Location of something physically within a container, including locations.

Direct Supertypes

Containment

Association Ends

physically contains : Spatial Entity [*]

A physical entity contained by another physical entity.{Transitive}

physically within : Physical Container [*]

Physical container in which the subject physical entity is contained.{transitive}

al

1.6.10 Association Class Recieving Container <<Relationship>>

Relationship between a fill action and the container that is filled with something.

Figure 14. Augmented Container

Direct Supertypes

Effect

Association Ends

put into : Container [1]

Container that receives something by a fill action.

filled by : Add To Container Event [*]

Action that puts or moves something into a container or location.

1.6.11 Class Relocation

The transfer (send/receive) of something between containers or locations. As an Event, the move may be initiated by an actor different from the sender/receiver or may not be caused by an actor.
Examples include movement of a vehicle from one location to another or movement of supplies between warehouses.

Direct Supertypes

Add To Container Event, Removal Event

1.6.12 Class Removal Event

An event that removes things from containers or locations. This results in the termination of a containment relationship.

Direct Supertypes

Containment Event

1.6.13 Association Class Supplying Container <<Relationship>>

Relationship between a removal action and the container it removes something from.

Figure 15. Depleted Container

Direct Supertypes

Effect

Association Ends

remove from : Container [1]

Container (or location) that contained something that is removed.

removed by : Removal Event [0..*]

Action that removes something from a container or location.

1.7 Concept Library::Control

Concepts relating to actor's control over resources.

1.7.1 Diagram: Control

Figure 16. Control

This diagram shows that control is a kind of ability and that possession, ownership and automated control are kinds of control. For each kind of control there are specific kinds of roles involved.

1.7.2 Diagram: Control Authority

Figure 17. Control Authority

1.7.3 Diagram: Custody

Figure 18. Custody

Custody provides a general framework for tracking the control, provenance and life cycle of items and information such that the history, trust and provenance may be ascertained. Custody provides for secure supply chains in that the life-cycle or items, information and their parts may be traced.
Custody is the relation between a Custodian and a Managed Entity (Something for which the provenance is interesting). Trust in a managed entity may also be influenced by the actors that have a capability to impact the resource.

1.7.4 Class Authority <<Role>>

An actor with authority over resources such that it can assert policy or behavior.
[ISO 15779:2011] organization, office, or individual responsible for approving equipment, installations or procedures

Direct Supertypes

Controlling Actor, Stakeholder

1.7.5 Association Class Control <<Relationship>>

The use or influence of an actor over an entity. This includes subtypes of control representing possession, ownership, leadership. and custody.
[FIBO] Control: the possession by a party, direct or indirect, of the power to direct or cause the direction of the management and policies of a thing, whether through the ownership of voting shares, by contract, or otherwise.

Figure 19. Control Detail

Direct Supertypes

Ability

Association Ends

has control over : Controlled Entity [1..*]

Entity which an actor controls in some way - by authority or by possession, by force, etc.
[FIBO] controls

is controlled by : Controlling Actor [1..*]

Actor which controls a controlled entity in some way.The nature of control may be refined in subtypes.
[FIBO] isControlledBy

1.7.6 Class Controlled Entity <<Role>>

Role of an entity that is controlled by a controlling actor.
[FIBO] ControlledThing: thing over which some party exercises some form of control in some context

Direct Supertypes

Actual Entity, Resource

1.7.7 Class Controlling Actor <<Role>>

Role of an actor that asserts control over any entity.
[FIBO] ContollingParty: Party which exercises some form of control in some context.

Direct Supertypes

Actor

1.7.8 Class Custodian <<Role>>

An actor who has responsibility for or looks after some managed entity. A Custodian <has custody of> a managed entity via the Custody relation.

Direct Supertypes

Controlling Actor

1.7.9 Association Class Custody <<Relationship>>

The act of a custodian protecting or taking care of a managed entity.

Direct Supertypes

Control

Association Ends

has custody of : Managed Entity [*]

The entity a custodian has custody of.

in the custody of : Custodian [*]

The custodian of a managed entity.

1.7.10 Class Leader <<Role>>

A person who leads or commands a group, organization, or country.

Direct Supertypes

Authority, Controlling Actor, Person

1.7.11 Association Class Leadership <<Relationship>>

An person leading or governing an organization.

Direct Supertypes

Control, Membership, Subject to Authority

Association Ends

leads : Organization [1..*]

An organization a person leads.

has leader : Leader [*]

A person leading or directing an organization.

1.7.12 Class Managed Entity <<Role>>

Any entity for which the custody of or access to the entity is managed such that it can be trusted or protected. A managed entity is in the custody of a custodian via the Custody relation.

Direct Supertypes

Controlled Entity

1.7.13 Class Owner <<Role>>

Role of an actor that owns property.
[FIBO] Owner: A party in the ownership role; one that owns something. The thing owned is an Asset to that Party.

Direct Supertypes

Controlling Actor

1.7.14 Association Class Ownership Control <<Relationship>>

Relationship defining the ownership of property by an owner.
[FIBO] Ownership: Ownership is the context in which some Party is said to own some Independent Thing. The Party is defined as such due to its being the owning party to that Thing.

Direct Supertypes

Control

Association Ends

owns : Property [1..*]

Property owned by an owner.
[FIBO] owns: to have (something) as one's own, possess

owned by : Owner [1..*]

Owner of an entity as property.
[FIBO] isOwnedBy: identifies the party that owns the asset.
[NIEM] ItemOwner

1.7.15 Association Class Possession <<Relationship>>

A relationship defining the physical possession of an item.

Direct Supertypes

Control

Association Ends

possesses : Controlled Entity [*]

A controlled entity in the physical possession of a controlling actor.

is possessed by : Controlling Actor [*]

The actor that possesses the subject controlled entity.
[NIEM] ItemPossessor

1.7.16 Class Property <<Role>>

Role of an entity which has an owner.
[FIBO] Asset: A thing held by some party and having some value.

Direct Supertypes

Controlled Entity

1.7.17 Association Class Subject to Authority <<Relationship>>

The relationship between an authority and what it has authority over.

Direct Supertypes

Control

Association Ends

has authority over : Controlled Entity [*]

Resource an authority has authority over - may influence in some by setting policy or defining requirements.
[FIBO] governs

is subject to authority : Authority [*]

The authority that has some control over a resource.
[FIBO] isGovernedBy

1.8 Concept Library::Credentials

Concepts relating to identity and credential management.

1.8.1 Diagram: Credentials and Managed Identifiers

Figure 20. Credentials and Managed Identifiers

1.8.2 Association Actor Identifier of Credential

Association Ends

credential identifier : Managed Actor Identifier

Managed identifier used by a credential to identify the credential and/or actor.

referenced by credential : Credential [*]

A credential that use the subject managed identifier to identify the credential or actor.

1.8.3 Association Class Attest to Ability <<Relationship>>

Relationship between a credential and the abilities it attests to. Note that ability is a relationship between an actor and a resource, so the credentialed individual is the actor in this relationship, the actor is identified by an identifier referenced by the credential.

Figure 21. Attest to Ability

Direct Supertypes

Impact, Relationship

Association Ends

attests to : Ability [0..*]

A statement of a credentialed capability - that an actor has the stated ability to utilize a resource.

has credential : Credential [0..*]

A physical or logical record of an assertion that a particular actor has a particular ability.

1.8.4 Class Credential

A credential is an attestation of qualification, competence, or authority issued to an individual by a third party with a relevant or de-facto authority or assumed competence to do so. Credentials can be physical (a house key), documents (a certificate) or virtual (a PKI key) and may be valid for a specific time frame and in specific context.

Direct Supertypes

Actual Entity, Controlled Entity

1.8.5 Class Identity Provider <<Role>>

Role of an organization that validates identity and issues curated identifiers and credentials for entities.

Direct Supertypes

Authority, Namespace, Organization

1.8.6 Association Class Issue Credential <<Relationship>>

Issuance of and the assertion of the validity of a credential by an identity provider.

Figure 22. Issue Credential

Direct Supertypes

Subject to Authority

Association Ends

issues credential : Credential [*]

Credential issued by an identity provider. By issuing the credential the identity provider is asserting the validity of the credential.

issued by : Identity Provider [1]

The identity provider issuing a credential By issuing the credential the identity provider is asserting the validity of the credential.

1.8.7 Class Managed Actor Identifier <<Value>>

An identifier managed by an identity provider who asserts the validity of the identifier. This includes technical/cyber identities as well as traditional identifiers such as passport numbers and corporate IDs. Identities can also be provided for systems, such as a SSL certificate.

Direct Supertypes

Unique Identifier

1.8.8 Association Class Valid for Time Interval <<Relationship>>

Relationship describing the time interval for which a credential is valid.

Figure 23. Valid for Time Interval

Direct Supertypes

Entity Exists for Interval

Association Ends

valid for time : Time Interval [1]

Time interval over which a credential is valid (can be used to provide evidence for an ability).

timeframe for : Credential [0..*]

Credential that is validated within the <valid for time> interval.

1.9 Concept Library::Cyber

The Cyber package defines instances and subtypes of generic concepts specific to Cyber - computers, software and networks.

1.9.1 Diagram: Cyber

Figure 24. Cyber

1.9.2 Diagram: Cyber Platforms

Figure 25. Cyber Platforms

1.9.3 Diagram: Cyber Resource

Figure 26. Cyber Resource

1.9.4 Association Class Ability To Execute Software <<Relationship>>

Relationship between software and platforms that can execute that software. Platforms include specific computers , types of computers and software.

Figure 27. Ability To Execte Software

Direct Supertypes

Ability, Cyber Resource

Association Ends

executed by : Execution Platform [0..*]

A computer platform (software, computer type or specific computer) able to execute specific software.

can execute : Software [1..*]

Software a computer system, other software or computer type is able to execute.

1.9.5 Association Class Automated Capability <<Relationship>>

Capability of a specific computer system to automate activities.

Figure 28. Information Processing Capability

Direct Supertypes

Cyber Resource

Association Ends

automates : Activity [*]

Activities a computer may automate.

automated by : Computer System [0..*]

Computer system capable of automating an activity.

1.9.6 Association Class Automated Control <<Relationship>>

Control of an entity by an automated control system - a computer and related software.

Figure 29. Automated Control

Direct Supertypes

Control

Association Ends

automates : Automated Entity [1..*]

Entity that is controlled by a computer system.

has control system : Computer Control System [1..*]

Control system for an automated entity. e.g., an automated machine or facility.

1.9.7 Class Automated Entity <<Role>>

Any actual entity all or partially controlled by automation.

Direct Supertypes

Controlled Entity

1.9.8 Class Automation Type

A categorization of computers and software across any dimension - chip type, operating system, language, manufacturer, virtual machine, etc. The automation type may be used to establish software compatibilities and vulnerabilities. Note that any automation may be categorized by multiple automation types.

Direct Supertypes

Cyber Resource, Entity Type, Execution Platform

1.9.9 Class Automaton

A machine or group of machines (most often a computer system, robot, or computerized swarm combined with software), or software that can perform actions in accordance with a process without another actor directing each step of the process. Distinguished from simple tools which facilitate an actor performing a process but have no innate ability to follow such a process.

Automation is distinguished from "legal entity" and "stakeholder", roles of some actors which indicates the ability to enter into legally binding agreements or have objectives (at this time no Automatons are legal entities or stakeholders but the model does not preclude the possibility).

Direct Supertypes

Actor, System

1.9.10 Class Communicating Device

A device able to communicate or facilitate communications across a network.

Direct Supertypes

Contactable, Cyber Resource, Device

1.9.11 Class Communications Link

A physical or virtual link between communications devices allowing them to communicate.

Direct Supertypes

Communicating Device, Communications Network

1.9.12 Class Communications Network

A physical or electronic system intended to facilitate communications between entities. Includes communications channels, computer networks, physical mail and RF networks.

Direct Supertypes

Actual Entity, Composite, Namespace

Attributes

◇ provides security level : Communications Security Level

The level of security asserted for the subject communications network.

1.9.13 Class Computer Control System <<Role>>

A computer control system is a device, or set of devices, that manages, commands, directs or regulates the behavior of other devices or systems. Industrial control systems are used in industrial production for controlling equipment or machines.

Direct Supertypes

Computer System, Controlling Actor

1.9.14 Class Computer System

An identifiable and physical computer system that acts as an automaton agent performing processes. [ISO/IEC 10514-1:1996] The combination of hardware and, optionally, firmware and software (e.g. operating system) that enables the execution of software.

Direct Supertypes

Automaton, Communicating Device, Container, Cyber Resource, Execution Platform

Attributes

network address : Technical Identifier

Electronic address which allows communication with a computer system as a node on a network.

1.9.15 Class Cyber Resource <<Union>>

Resources that, together, make up information systems capabilities and may be vulnerable to attack or used in an attack.

1.9.16 Class Execution Platform <<Union>>

Computer hardware or software that provides the capability of executing software. This includes processors, operating systems and virtual machines.

Direct Supertypes

Cyber Resource

1.9.17 Association Class Information In Computer <<Relationship>>

Relationship defining information stored in a computer system.

Figure 30. Information In Computer

Direct Supertypes

Containment

Association Ends

stores information : Information Object [0..*]

Information stored in a computer.

is stored in : Computer System [*]

System storing an information object.

1.9.18 Association Class Node of a Network <<Relationship>>

Relationship between a network and nodes that may communicate on that network.

Figure 31. Node Of A Network

Direct Supertypes

Parthood

Association Ends

communicates via : Communications Network [*]

The network used by a communications device to send and receive information.

has nodes : Communicating Device [1..*]

The communicating nodes of a communications network. Nodes are able to communicate with each other across the network.

1.9.19 Class Software

Programs and other operating information used by a computer to control its function through the definition of a process.

Direct Supertypes

Automaton, Cyber Resource, Execution Platform, Information Object, Tool

1.10 Concept Library::Enterprises

In a generic sense, an enterprise is any organization or collection of organizations that has a common set of goals and/or a single bottom line. An enterprise, by that definition, can encompass a Military Department, DoD as a whole, a division within an organization, an organization in a single location, or a chain of geographically distant organizations linked by a common management or purpose. An enterprise today is often thought of as an extended enterprise where partners, suppliers, customers, along with their activities and supporting systems, are included in the Architectural Description. [DoDAF 2.0] section 51, Defining the Enterprise
The concept of enterprise builds on the concept of a system.

1.10.1 Diagram: Enterprise

Figure 32. Enterprise

1.10.2 Class Enterprise

An enterprise is a stakeholder organization, organized as a system, with a mission, members, and authority over resources to accomplish its mission(s). An enterprise provides context for operations and analysis. An enterprise may have parts - its divisions or departments.
[BMM] Organizational Unit:

Direct Supertypes

Authority, Organization, System

1.11 Concept Library::Entities

The foundation library provides fundamental concepts that apply to most domains or areas of concern. These fundamental concepts are specialized, combined and related for more specific concerns, such as risk management.
These fundamental concepts provide for links between domains, systems, organizations, cultures and stakeholders.
Unless stated otherwise, these concepts are intended to be mixed together to fully describe something in the "real world".
The foundation builds on the conceptual model defined in [SIMF] (Semantic Information Modeling for Federation), which is included by reference. By using SIMF we avoid redundant definitions of concepts and also allow for easy extension of threat/risk by using the SIMF modeling capabilities.
Note that something may be classified by any number of types (e.g., a transfer of custody that is an actual situation that happened in the past).

1.11.1 Diagram: Identifiable Entity

Figure 33. Identifiable Entity

The above is a summary diagram intended to show significant classes used in the concept library.

1.11.2 Diagram: Identifiable Entity Relationships

Figure 34. Identifiable Entity Relationships

1.11.3 Association Class Impact <<Relationship>>

Relationship between some entity and another on which it has some kind of impact or effect.

Figure 35. Impact

Direct Supertypes

Temporally Related

Association Ends

impacts : Entity [*]

Entity that the subject entity impacts in any way.

impacted by : Entity [*]

Entity that is impacted by another in any way..

Attributes

likelihood : Probability Metric

Metric representing the probability that the <negated by> situation will prevent the <negates> situation

1.11.4 Association Class Temporally Related <<Relationship>>

Temporally Related defines relationships as first-class entities that are "situations" that may have conditions, context or a time frame. "Related" is the implicit supertype of all entity relationships such that any relationship can be traced between entities. Note that the generalization to Temporally Related may not be shown on diagrams.

Note that in UML relationships that are "definitional" and not expected to be time or contextually dependent are shown as regular UML associations where as potentially contextual or time-bound relationships are shown as association classes. This is a notational convention and does not have semantic intent to avoid early commitment to such considerations.

Figure 36. Related

Direct Supertypes

Relationship, Temporal Entity

Association Ends

relates to : Entity [*]

A generic relationship to capture arbitrary relationships that do not have more specific meaning. <relates> is the implicit supertype of all relationships between entities (not including metadata). Note that to remove diagram clutter, subsets of "relates" may not show the subset on all diagrams.

related from : Entity [*]

1.12 Concept Library::Identifiers and Coordinates

Identification connects identifiers with the entity they identify. Identifiers are values, that is they are immutable "data". An entity may be assigned different identifiers over time and may have many at any one time.
The base Identifier class (defined in SIMF) identifies a set of entities and is not assumed to be unique or to identify only one entity. For example, a name is an identifier but many people could have the same name and a person could have multiple names. Identifiers and the relation to an entity may be contextual.
Identifiers should not be confused with Identity, which is an abstracton of individuality that provide the foundation for identifiers.
The class of Unique Identifiers is more typical of I.T. systems and managed identifiers, such as driver's license numbers. A unique identifier is assumed unique within exactly one Namespace.
Subtypes of Identifier may be specific to identifying particular kinds of entities. For example, a Location Identifier (such as an address or GPS coordinate) is specific to identifying locations.
Note that as with all concepts, identifiers are independent of representation. Since names are so often textual, we also define a representation of names - "Textual Name".
The generic library of identifiers extends the SIMF identifiers with the concepts of coordinates.

1.12.1 Diagram: Identifiers

Figure 37. Identifiers

1.12.2 Class Coordinate <<Value>>

Any set of magnitudes that serve to define the position of a point, line, or the like, by reference to a coordinate system.

Direct Supertypes

Unique Identifier

1.12.3 Class Coordinate System

A reference system for a coordinate. e.g., WGS-84.
[OGC] Set of mathematical rules for specifying how coordinates are to be assigned to points.

Direct Supertypes

Namespace

1.12.4 Association System of Coordinate

Relationship between a coordinate and a system of coordinates that defines how the coordinate is to be interpreted.

Association Ends

system of coordinate : Coordinate System [1]

The set of rules and reference points used for interpreting a coordinate.

coordinate within system : Coordinate [*]

A particular coordinate identifying an entity within the scope of a coordinate system.

1.13 Concept Library::Information

Concepts relating to information (including data) about entities.

1.13.1 Diagram: Information Action

Figure 38. Information Action

1.13.2 Diagram: Information Objects

Figure 39. Information Objects

1.13.3 Diagram: Metadata

Figure 40. Metadata

1.13.4 Class Add Information

An action to add information to a repository or information structure.

Direct Supertypes

Add To Container Event, Information Action

1.13.5 Class Atomic Information Object

An atomic information object is a managed piece of information composed entirely of a single value. e.g. an MP3 representation of a song. Note that different viewpoints and abstractions may not agree on what is "atomic". e.g. a music editor may consider a song a series of samples.

Direct Supertypes

Information Object

Attributes

value : Value [1]

Atomic value associated with an atomic information object.

1.13.6 Class Close Information

An action that removes information from visibility.

Direct Supertypes

Information Action

1.13.7 Class Confidence

A statement and measure of the confidence in any fact or set of facts about an entity.

Direct Supertypes

Metadata

Attributes

confidence metric : Confidence Metric

A metric reflecting confidence in an assertion condition or effect.

1.13.8 Association Confidence in Assertion

A relationship relating a degree of confidence with the topic or subject of that confidence. Confidence is metadata about statements in a model; the degree of belief in those statements.

Association Ends

confidence about : Entity [*]

Subject of confidence; the facts, situations or subject that confidence is being evaluated for.

has confidence : Confidence [0..1]

Confidence is a metric quantifying the belief that the facts asserted about the entity are true and valid.

1.13.9 Association Class Contained Information <<Relationship>>

Relationship connecting an information container with what it contains.

Figure 41. Contained Information

Direct Supertypes

Containment

Association Ends

contains information : Information Object [*]

An information object structurally contained in another.

is in information structure : Structured Information Object [*]

Structural containment of an information object within an information object.

1.13.10 Class Copy Information

An Information Action to duplicate information in another location or container.

Direct Supertypes

Add To Container Event, Information Action

1.13.11 Class Create Information

An action that creates information.

Direct Supertypes

Add To Container Event, Create, Information Action

1.13.12 Class Delete Information

An action to delete information, erase it or render it inaccessible.

Direct Supertypes

Destroy, Information Action

1.13.13 Class Document

A collection of information about a topic, frequently containing some analysis or summary, intended for use by a stakeholder.

Direct Supertypes

Structured Information Object

1.13.14 Class Information Action

An action that impacts information objects.

Figure 42. Information Action Detail

Direct Supertypes

Capability, Cyber Resource

1.13.15 Class Information Object

A representation of information, data, facts, assertions or statements about something.
As information may be copied while retaining its identity, the same information copied onto new physical media may be considered the same object. Where the individual representation of information is of concern another object should be used to represent the individual physical thing than holds the information.
As information is a resource it may depend on other resources.
[FIBO] Document
[NIEM] DocumentType

Direct Supertypes

Actual Entity, Cyber Resource, Entity, Information Resource, Resource

1.13.16 Class Information Repository

A resource in which information is stored and can then be retrieved.

Direct Supertypes

Structured Information Object

1.13.17 Class Information Resource <<Union>>

Information objects or types that can be manipulated by an information action.

1.13.18 Class Information Type

A categorization of information across any dimension - content, format, source, sensitivity, etc. e.g. a schema. The information type may be used to establish software capabilities and vulnerabilities.

Figure 43. Information Type

Direct Supertypes

Cyber Resource, Entity Type, Information Resource

1.13.19 Class Modify Information

Action to change information (for good or bad reasons).

Direct Supertypes

Information Action

1.13.20 Class Open Information

Action to gain visibility to some information, e.g., Open a file or an envelope.

Direct Supertypes

Information Action

1.13.21 Class Read Information

An action to read, access, or understand some information.

Direct Supertypes

Information Action

1.13.22 Class Remove Information

An action to remove information from some repository or composite information structure.

Direct Supertypes

Information Action, Removal Event

1.13.23 Class Structured Information Object

An information object that contains sub-elements. e.g., a "record".

Direct Supertypes

Container, Information Object

1.13.24 Class Transfer Information

The transfer of information from one information store to another.

Direct Supertypes

Add Information, Relocation, Remove Information

1.14 Concept Library::Locations

Concepts related to locations and places.

1.14.1 Diagram: Location

Figure 44. Location

1.14.2 Diagram: Location Identification

Figure 45. Location Identification

1.14.3 Association Address of Location

identification of a location by an address. Note that there are postal addresses that do not identify a location, so this relation is optional. However, most postal addresses do identify a location thus this relation is possible.

Association Ends

has address : Postal Address [*]

A postal address of a physical location.

address of : Physical Location [0..1]

Location identified by an address.

1.14.4 Association Coordinate of location

Relationship between a physical location and the coordinate that defines its position.

Association Ends

has coordinate : Spacial Coordinate [*]

A coordinate that identifies a location.

Coordinate of location : Physical Location

Coordinate of location based on coordinate system.

1.14.5 Association Designation of a Location

Relationship defining the location identified by a location identifier.

Association Ends

designates location : Physical Location [1]

The physical location identified or described by a location identifier..

has location designation : Location Identifier [0..*]

A description or identifier that designates a particular location.

1.14.6 Class Location ID <<Value>>

A code, ID or name for a physical location.

Direct Supertypes

Location Identifier, Text Identifier

1.14.7 Class Location Identifier <<Value>>

Any identifier able to uniquely identify a physical location
Syn. spatial reference - description of position in the real world [OGC]

Direct Supertypes

Unique Identifier

1.14.8 Association Class Physical Boundary <<Relationship>>

Boundary describing the topology of a location.

Figure 46. Physical Boundary

Direct Supertypes

Temporally Related

Association Ends

bounded by : Physical Location [0..*]

The edge points of a topology where each successive pair of features (as well as the first and last points), connected by lines, describes a boundary.

bounds topology : Topology [0..*]

A location identified by geographic boundaries.

1.14.9 Class Physical Location

A point or extent in physical space.

[NIEM] A geospatial location.
[FIBO] PhysicalLocation: A location in physical space

Direct Supertypes

Physical Feature

1.14.10 Class Physical Point

A dimensionless physical point in space or on the surface of the earth such as a corner or center point.

Direct Supertypes

Physical Location

1.14.11 Class Point On Earth <<Value>>

A point that defines a location on earth where the point is within the bounds of <designates location>.

Direct Supertypes

Spacial Coordinate

1.14.12 Association Reference Point

Reference point for a relative location

Association Ends

relative to : Physical Location [1]

Where the position of something is relative to a location, the reference location.

has relative location : Relative Coordinate [*]

1.14.13 Class Relative Coordinate <<Value>>

A coordinate described relative to another. e.g., 5 miles west of the empire state building.

Direct Supertypes

Spacial Coordinate

Attributes

distance : Length

Distance as part of a relative coordinate that, when combined with angle, identifies a point <relative to> another point.

direction : Angle

An angle as part of a coordinate.

altitude : Length

Measure of how much something is above <relative to> something else, usually the earth.

1.14.14 Class Spacial Coordinate <<Value>>

Any point that uniquely identifies a spacial location relative to a coordinate system.

One of a sequence of n numbers designating the position of a point in n-dimensional space [OGC]

Direct Supertypes

Coordinate, Location Identifier

1.14.15 Association Topological Region

Physical location described by a topology.

Association Ends

topology of : Physical Location [1]

Location described by a topology.

has toloplogy : Topology [0..1]

Topology that describes a physical location in terms of physical boundaries.

1.14.16 Class Topology

A record of a contiguous 1, 2 or 3 dimensioned area defined by geographic features and points.
[NIEM] AreaType

1.14.17 Class World Geodetic System <<Value>>

The World Geodetic System defines a reference frame for the earth, for use in geodesy and navigation. The latest revision is WGS 84 dating from 1984. [WGS-84]
[NIEM] Location2DGeospatialCoordinateType or Location3DGeospatialCoordinateType (With elevation)

Direct Supertypes

Point On Earth

Attributes

latitude : Angle
Latitude based on the prime meridian.
[FIBO] hasLatitude

longitude : Angle
Longitude based on the prime meridian.
[FIBO] hasLogitude

elevation : Length
Height above nominal sea level.

1.15 Concept Library::Objectives

Objectives captures how the intents of *stakeholder's* relate to the real-world *consequences* of *situations* that have or may happen. A *consequence* results in a *benefit* or *harm* to these *objectives* - generally related to a specific entity of value to the stakeholder.
As any situation may have multiple consequences, both benefits and harms, the net desirability of any situation to a stakeholder is calculated in the *Stakeholder Desirability Relation* by combining all of the related consequences.

1.15.1 Diagram: Objectives

Figure 47. Objectives

1.15.2 Class Benefit

A benefit is a consequence of a situation having positive desirability.

Direct Supertypes

Consequence

1.15.3 Class Consequence

A consequence or impact of the outcome of a situation affecting objectives of a stakeholder.
NOTE 1 An event can lead to a range of consequences.
NOTE 2 A consequence can be certain or uncertain and can have positive or negative effects on objectives.
NOTE 3 Consequences can be expressed qualitatively or quantitatively.
NOTE 4 Initial consequences can escalate through knock-on effects.
[ISO 73-2009]

Figure 48. Impact Detail

Direct Supertypes

Situation

Attributes

degree of affect : Metric
A metric for how much the consequence affects an objective - a measure of harm or benefit.

desirability : Metric
A metric describing the desirability of an impact. May be positive or negative where positive is desirable and negative is undesirable.

impact measure : Metric
A metric for impact where Impact = desirability * likelihood

1.15.4 Association Class Consequence of Situation <<Relationship>>

Impact of a situation - its affect on the objectives of stakeholders.

Figure 49. Consequence of Situation

Direct Supertypes

Cause and Effect

Association Ends

has consequence : Consequence [*]

A consequence of a situation that impacts the objectives of a stakeholder.

results from : Situation [1..*]

Situation causing an impact..

1.15.5 Class Desirability Assessment

Desirability measure of a situation. The context of the desirability can be the specific stakeholder desirability relation or a classification of a situation in a context, such as an undesirable situation.
Desirability may be computed by aggregating the impact of a situation on stakeholders but the specific calculation is not specified in the standard.
A positive desirability is an opportunity; a negative desirability is a danger.
When the stakeholder desirability of a situation to a stakeholder has net harm, that harm may be identified as a risk. S

Direct Supertypes

Assessment

Attributes

net desirability : Metric

A metric representing the aggregation of the impact of all consequences of a situation for a stakeholder. (net benefit - net risk)

net severity : Metric

The aggregation of the impact of all detriments (negative consequences) of a situation for a stakeholder.

net benefit : Metric

A metric representing the aggregation of the impact of all benefits (positive consequences) of a situation for a stakeholder.

net likelihood : Metric
The net sum of the likelihood of a situation or risk.

net risk : Metric
A metric representing the aggregation of risk metrics computed as likelihood*impact.

1.15.6 Class Means <<Role>>

[BMM] A means represents any device, capability, regime, technique, restriction, agency, instrument, or method that may be called upon, activated, or enforced to achieve Ends.

Direct Supertypes

Entity

1.15.7 Association Class Means To End <<Relationship>>

The relation between a means and an objective such that the means supports the objective.
[BMM] means is impacted by influencer

Figure 50. Means To End

Direct Supertypes

Impact

Association Ends

achieves : Objective [*]

Objectives supported by a means.

realized by : Means [*]

Means that serves to meet an objective.

1.15.8 Class Objective

An aim or goal that a stakeholder intends to attain or accomplish; purpose; goal; target.
[BMM] End: something that is to be accomplished.
[BMM] Objective: An Objective is a statement of an attainable, time-targeted, and measurable target that the enterprise seeks to meet in order to achieve its Goals.
[FIBO] Objective

Figure 51. Objective Detail

Direct Supertypes

Entity

1.15.9 Association Class Objective of Stakeholder <<Relationship>>

Relationship between a stakeholder and their objectives.

Figure 52. Objective of Stakeholder

Direct Supertypes

Temporally Related

Association Ends

has objective : Objective [*]

An objective that a stakeholder intends to retain or achieve.

objective of : Stakeholder [1..*]

A stakeholder having an objective they intend to attain or retain.

Attributes

importance : Metric

A metric of importance of an objective to a stakeholder.

1.15.10 Class Opportunity

An opportunity is any potential future situation having beneficial consequences.

[BMM] Opportunity: This category of Assessment indicates that some Influencer can have a favorable impact on the organization's employment of Means or achievement of Ends. For example, the bankruptcy of Pizza Company's major competitor in Region-Y is assessed to be an Opportunity in its Goal "To increase market share."

Figure 53. Opportunity Detail

Direct Supertypes

Potential Situation, Situation

1.15.11 Class Stakeholder <<Role>>

A stakeholder is a responsible performer having objectives and promoting the means for achieving those objectives.

Direct Supertypes

Social Agent

1.15.12 Association Class Stakeholder Desirability <<Relationship>>

A relationship representing the net desirability of a situation for a stakeholder.
Note: Stakeholder desirability is expected to be computed based on aggregating the impact of a situation for a stakeholder. However, the algorithm for this aggregation is not specified in the standard.

Figure 54. Stakeholder Desirability

Direct Supertypes

Desirability Assessment, Temporally Related

Association Ends

desirability of : Situation [*]

The situation evaluated in terms of its desirability for a stakeholder.

desirability for : Stakeholder [0..*]

A stakeholder for which desirability of a situation is evaluated.

1.16 Concept Library::Observations

Observations are acts where an observer notes some entity (including situations and individuals') that are observed in a situation.

1.16.1 Diagram: Measurement

Figure 55. Measurement

1.16.2 Diagram: Observations

Figure 56. Observations

1.16.3 Association Context of Observation

Context of an observations - e.g. the physical place, situation or timeframe in which the observation was made.

Direct Supertypes

Extent of Context

Association Ends

observed in context : Entity [*]

The context of an observation, what environment, location, timeframe or system is being observed. e.g. A man with a gun observed in an airport. The man is <observed in context> of the airport.

context of observation : Observation [*]

Observations made in the context of the subject entity. E.G. Sightings within an airport.

1.16.4 Association Class Measurement <<Relationship>>

A measurement is an observation made by <observed by> that <observes> the value of a characteristic for a particular entity, which is the Characteristic Binding the observer <observes>. The characteristic binding binds a particular value, e.g. 2 meters, with a particular characteristic, e.g. height, of a particular individual, e.g. John Smith.
As a characteristic binding is a temporal entity it has a time and context which may be different from the time and context of the measurement. e.g. The nurse "Sue" took the patients (Joe) weight measurement (Characteristic Binding - Joe <has weight> 94 KG) on 2/5/2010 at 9:31AM which was recorded as the patents current weight for 90 days.

Direct Supertypes

Observation

Association Ends

: Characteristic Binding [*]

: Observer [0..*]

1.16.5 Association Class Observation <<Relationship>>

An observation is an actual (not possible) activity of the <observed by> observer where <observes> has been noticed or perceived as being significant. The observation is <observed in context> such as a place or condition. The observation <uses> any number of observation tools.
Example: Sam, a driver and Observer, notices <observes> a Deer <observed in context> of the road on which he is driving.

Figure 57. Observation

Direct Supertypes

Actual Activity, Impact, Involvement

Association Ends

observes : Entity [1..*]

Entity observed by an observer making an observation.

observed by : Observer [0..*]

Observations of an entity by an observer.

Attributes

number observed : Numeric Value

The number of individual observations aggregated into a single observation. The <observes> entity will likely be a type. E.G. Sue saw 5 birds.

1.16.6 Class Observation Tool <<Role>>

A tools that assists in observations. e.g. a wireless microphone is used to observe a conversation.

Direct Supertypes

Tool

1.16.7 Class Observer <<Role>>

Role of an actor that can or has observed something

Direct Supertypes

Actor

1.17 Concept Library::Occurrences

Events are things that occur in time, impacting the things involved in those events. Events include actual events as well as patterns of events that describe a process.

1.17.1 Diagram: Occurrences

Figure 58. Occurrences

1.17.2 Class Activity

An Event performed by one or more actors intended to meet a need.
[UAF] Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.

Direct Supertypes

Occurrence, Resource

1.17.3 Class Actor

An entity capable of behavior - performing an activity or process.

[IDEAS] Agent: Something capable of action.
[FIBO] AutonomousAgent: An agent is an autonomous individual that can adapt to and interact with its environment.

Figure 59. Actor Detail

Direct Supertypes

Actual Entity, Contactable

1.17.4 Class Actual Activity <<Intersection>>

A specific, actual, activity that has or may happen.

Direct Supertypes

Activity, Actual Occurrence

1.17.5 Class Actual Occurrence <<Intersection>>

A specific individual occurrence that has happened, is happening or may happen.

[FIBO] Occurrence: An Occurrence is a happening of an OccurrenceKind. Each Occur-rence has a DateTimeStamp, which identifies when the Occurrence happened, and a Location (possibly virtual), that identifies where the Occurrence happened.

Direct Supertypes

Actual Situation, Occurrence

1.17.6 Class Occurrence

An Occurrence is something that happens (a.k.a. occurs). A dynamic situation (past, present or future) composed of a set of things changing over a period of time. e.g., a rock falling.
Events are not limited in their timeframe. Events can have long or short timeframes, from an instant to infinity and beyond.
An Event that is "performed by" an actor is considered an activity.

[DOLCE] Perdurant

[BFO] Event: perdurant that is related to exactly two states (its pre-state and its post-state).
An event is related to the states before and after it has happened.

[NIEM] ActivityType

Direct Supertypes

Situation

1.17.7 Association Class Output <<Relationship>>

Outputs from a process or actual event - the things or situations it creates.

Figure 60. Output

Direct Supertypes

Effect

Association Ends

produces : Entity [*]

Resources produced by a process or actual event

produced by : Occurrence [*]

Occurrences which produce an entity.

1.17.8 Association Class Performance <<Relationship>>

Performance is the act of an actor as the driving force in the execution of an activity. Related to "Capability" as the ability to perform.

[DOLCE] Subtype of Participation

Figure 61. Performance

Direct Supertypes

Impact, Involvement, Occurrence

Association Ends

performs : Activity [1..*]

An activity performed (executed or enacted) by an actor.

performed by : Performer [1..*]

The actor which is the performer of an activity.

1.17.9 Association Class Usage <<Relationship>>

Inputs to a process or actual event - what it uses

Figure 62. Usage

Direct Supertypes

Effect

Association Ends

uses : Resource [0..*]

A resources used by a process or actual event.

used by : Occurrence [*]

A process or actual occurrence that is used by a resource for the resource to fulfill its function.

1.18 Concept Library::Organizations

An Organization is group of persons and/or other actors and resources organized for some end or work. Subtypes of organizations include governments and corporations.

1.18.1 Diagram: Organization

Figure 63. Organization

1.18.2 Association Class Membership <<Relationship>>

Relationship representing the participation of an actor in an organization. Subtypes of membership may provide more explicit membership kinds.

Figure 64. Membership

Direct Supertypes

Associated Actor

Association Ends

has member : Member [*]

An assertion of membership in an organization.
[FIBO] hasMember

member of : Organization [*]

Organization a performer belongs to.
[FIBO] memberOf

1.18.3 Class Mission Objective

A core objective of an enterprise.

[BMM] A Mission indicates the ongoing operational activity of the enterprise. The Mission describes what the business is or will be doing on a day-to-day basis.

Direct Supertypes

Objective

1.18.4 Class Organization

An Organization is a group of persons and/or other actors and resources organized for some end or work
[FIBO] Organization: a social unit of people, systematically structured and managed to meet a need or pursue collective goals on a continuing basis.
[NIEM] OrganizationType
[DOLCE] Society

Direct Supertypes

Controlled Entity, Social Agent, Stakeholder

1.18.5 Class Organizational Unit <<Role>>

Organizational unit is a role that encompasses subdivisions, departments, subsidiaries and other organizational parts of an organization.
[BMM] organization unit: An administrative or functional unit within an organization structure.

Direct Supertypes

Controlled Entity, Organization

1.18.6 Class Parent Organization <<Role>>

Organization with component parts such as divisions and departments.

Direct Supertypes

Composite, Controlling Actor, Organization

1.18.7 Association Class Part of Organization <<Relationship>>

Relationship between an organization and its organizational units - the sub-organizations such as departments and subsidiaries.

Figure 65. Part Of Organization

Direct Supertypes

Control, Parthood

Association Ends

has parent organization : Parent Organization [1..*]

The parent (controlling organization) of another organization.

has child organization : Organizational Unit [1..*]

An organization constituted as a component of another organization.

1.18.8 Class Program

A set of projects, activities, or services of an organization that are intended to meet a need.
[NIEM] ProgramType

Direct Supertypes

Activity

1.19 Concept Library::Organizations::Corporations

1.19.1 Diagram: Corporations

Figure 66. Corporations

1.19.2 Class Incorporated Organization

An organization recognized by and incorporated by a recognized government.
[FIBO] FormalOrganization: an organization that is recognized in some legal jurisdiction, with associated rights and responsibilities

Direct Supertypes

Organization

1.19.3 Association Class Incorporation <<Relationship>>

Act by which individuals' are voluntarily united into a new entity through the creation of an artificial, intangible, and legal person called a corporation.

Figure 67. Incorporation

Direct Supertypes

Associated Actor, Subject to Authority

Association Ends

incorporated by : Geopolitical Entity [1]

Geopolitical entity incorporating an organization.

incorporates : Incorporated Organization [*]

An organization incorporated by a government.

1.20 Concept Library::Organizations::Geopolitical Organizations

1.20.1 Diagram: Geopolitical Entities

Figure 68. Geopolitical Entities

1.20.2 Class Country

A nation with its own government, occupying a particular territory (not necessarily contiguous).
[FIBO] Country: A self-governing geopolitical unit that is recognized as a country by the United Nations (more specific concept as being recognized by the U.N.)

Direct Supertypes

Geopolitical Entity

1.20.3 Class Country ID <<Value>>

A code, ID, or name for a country.

[ISO 1087] country identifier: information in a terminological entry (3.8.2) which indicates the name of a geographical region where the designation (3.4.1) is used

Direct Supertypes

Geopolitical ID

1.20.4 Class Geopolitical Entity

An organization which is the governing body of a nation, state, tribe or community.
[FIBO] GeopoliticalEntity (In FIBO this is a subclass of Physical Location. In Threat/risk Geopolitical Region is a role of a physical location. FIBO combines Geopolitical Entity with Geopolitical Region).

Direct Supertypes

Authority, Organization

1.20.5 Class Geopolitical ID <<Value>>

A code, ID or administered name for a geopolitical entity with governmental authority e.g., city, state, county, tribe.

Direct Supertypes

Text Identifier

1.20.6 Class Geopolitical Region <<Role>>

A physical location governed by a geopolitical entity.
[FIBO] GeopoliticalEntity (In FIBO this is a subclass of Physical Location. In Threat/risk Geopolitical Region is a role of a physical location. FIBO combines Geopolitical Entity with Geopolitical Region.
[NIEM] LocaleType

Direct Supertypes

Physical Location

1.20.7 Association Class Governing Authority <<Relationship>>

Relationship representing authority over a region.

Direct Supertypes

Associated Actor, Subject to Authority

Association Ends

governs region : Geopolitical Region [1..*]

Region governed by a geopolitical entity.

is governed by : Geopolitical Entity [*]

A governing authority for a region.

1.21 Concept Library::Parthood

1.21.1 Diagram: Parthood

Figure 69. Parthood

1.21.2 Class Composite <<Role>>

Direct Supertypes

Context, Entity

1.21.3 Class Part <<Role>>

Direct Supertypes

Entity

1.21.4 Association Class Parthood <<Relationship>>

Relationship defining one thing as a part of another. More specific concepts of parthood and mereology (the study of parts and their relations) may subtype Parthood.

[IDEAS] wholePart: The whole-part pattern establishes a relationship between individual elements, asserting that one Object element is composed of the other element.

[IDEAS] A couple that asserts one (part) Individual is part of another (whole) Individual.

[ISO 1087] partitive relation: part-whole relation relation between two concepts (3.2.1) where one of the concepts constitutes the whole and the other concept a part of that whole

[DOLCE] Parthood

Figure 70. Parthood

Direct Supertypes

Temporally Related

Association Ends

has part : Part [1..*]

Entity that is a part of the subject entity (the whole) such that the part is essential to the whole. This is a general concept of part and does not assume exclusivity of partness or total part inclusion.

[FIBO] hasPart

[ISO 1087] partitive concept: concept (3.2.1) in a partitive relation (3.2.22) viewed as one of the parts making up the whole

is part of : Composite [*]

Composite entity (whole) of which this entity is a part.

[FIBO] isPartOf

[ISO 1087] comprehensive concept: concept (3.2.1) in a partitive relation (3.2.22) viewed as the whole

1.22 Concept Library::Permissions

Concepts relating to the permission an actor has to perform some process.

1.22.1 Diagram: Permission

Figure 71. Permission

1.22.2 Association Class Permission <<Relationship>>

Permission is a relationship representing authorization granted by an authority to an actor to perform a kind of activity. THe activity may be either an "Actual Activity" or a kind of activity - a Modus Operandi.

Direct Supertypes

Ability, Controlled Entity

Association Ends

has permission to perform : Activity [*]

Activity the actor has permission to perform.

may be performed by : Actor [*]

Actors that have permission to perform the subject activity.

Attributes

revokes permission : Boolean

Inverts or removes the permission asserted by the permission relationship.

1.23 Concept Library::Persons

This person module defines foundation concepts of people such as their location and name. More specific person attributes may augment this specification.

1.23.1 Diagram: Person

Figure 72. Person

1.23.2 Diagram: Person Identifiers

Figure 73. Person Identifiers

1.23.3 Diagram: Person Name Representations

Figure 74. Person Name Representations

1.23.4 Class Access Identifier <<Value>>

An term, data value or other sign that identifies a person for access to a resource.

Direct Supertypes

Managed Person Identifier

1.23.5 Class Financial Identifier <<Value>>

An identifier for purposes of making financial transactions, such as a credit card number or bank account.

Direct Supertypes

Managed Social Agent Identifier

1.23.6 Class Managed Person Identifier <<Value>>

An identifier for a person managed by some identity provider who asserts the validity of the identifier, frequently but not always a government organization.

Direct Supertypes

Managed Social Agent Identifier

1.23.7 Class Passport Identifier <<Value>>

[NIEM] PersonPassportIdentification (property): An identification of a passport issued to a person.

Direct Supertypes

Managed Person Identifier

1.23.8 Class Person

An individual human being.
[FIBO] Person
[NIEM] PersonType
[DOLCE] (Subtype of) Agentive Physical Object

Direct Supertypes

Animal, Social Agent

1.23.9 Association Class Person at location <<Relationship>>

A relationship representing the location of a person at a particular time.
[NIEM] PersonLocationAssociationType

Figure 75. Person at location

Direct Supertypes

Physical Containment

Association Ends

location of person : Physical Location [*]

Location of the subject person.

[FIBO] isSituatedAt (mode general concept)

located person : Person [*]

A person who is at a location.

1.23.10 Class Person Name <<Value>>

Text identifying a person by a recognized name.
[FIBO] hasFullLegalName (More specific concept)
[NIEM] PersonNameType

Direct Supertypes

Name

1.23.11 Class Person Structured Name <<Value>>

A full name of a person in a structured form.
Note: Conversion between structured and textual names is provided by the implementation and is not defined in this specification.

Direct Supertypes

Person Name

Attributes

title : Text Identifier [0..*]
[NIEM] PersonNamePrefixText: A title or honorific used by a person.

salutation : Text Identifier [0..1]
[NIEM] PersonNameSalutationText: A formal sign or expression of greeting that is appropriate for a person.

name part : Text Identifier [1..*]
Parts of a person's name, e.g., surname, given name.
[FIBO] hasFamilyName

given name : Unique Text Identifier
[NIEM] PersonGivenName: A first name of a person.
[FIBO] hasGivenName

official given name : Unique Text Identifier
[NIEM] PersonOfficialGivenName: A name, out of possibly multiple given names, that a person selects to use as his or her official given name.

middle name : Unique Text Identifier
[NIEM] PersonMiddleName: A middle name of a person.

surname prefix : Unique Text Identifier
[NIEM] PursonSurNamePrefix: A prefix that precedes this person's family name such as Van, Von.

surname : Unique Text Identifier
[NIEM] PersonSurName: A last name or family name of a person.
[FIBO] hasSurname

name suffix : Unique Text Identifier
[NIEM] PersonNameSuffixText: A term appended after the family name that qualifies the name.

1.23.12 Association Class Residency <<Relationship>>

A residence of a person - where they live.

[NIEM] PersonResidenceAssociationType

[FIBO] Residence: Note that residence is not the same as domicile, as a person or organization can have many transient residences but only one legal domicile. The domicile of a formal organization is the address (location) where the establishment is maintained or where the governing power of the organization is exercised.

Figure 76. Residency

Direct Supertypes

Operating Location

Association Ends

resides at : Residence [*]

A residence of a person, where they live.

[FIBO] isDomiciledIn: identifies the permanent home or principal establishment of an individual or organization

has resident : Person [1..*]

A person living in a residence.

1.23.13 Class Social Security Number <<Value>>

[NIEM] PersonSSNIdentification (property): A unique identification reference to a living person; assigned by the United States Social Security Administration.

Direct Supertypes

Managed Person Identifier

1.24 Concept Library::Physical Entities

This package defines a hierarchy of physical entities and items. Items are inanimate material object as distinct from a living sentient being.

1.24.1 Diagram: Physical Entities

Figure 77. Physical Entities

1.24.2 Class Animal

Any member of the kingdom Animalia, comprising multicellular organisms that have a well-defined shape and usually limited growth, can move voluntarily, actively acquire food and digest it internally, and have sensory and nervous systems that allow them to respond rapidly to stimuli. A super type of "Person".

Figure 78. Animal Detail

Direct Supertypes

Actor, Physical Entity

Attributes

birth date : Time Point
The date an animal (including a person) was born, became an independent entity.
[FIBO] hasDateOfBirth

death date : Time Point
The date an animal (including a person) died, ceased to be living.

physical sex : Sex Kind
Sex of a living thing as indicated by essential physical characteristics, primarily genitalia.
[FIBO] hasGender

standing height : Length
The measurement from base to top or (of a standing person) from head to foot. "Current" is relative to the time frame of the defining context.

1.24.3 Class Conveyance

A device or system providing a means of physical transport from place to place.
[NIEM] ConveyanceType

Direct Supertypes

Device

1.24.4 Class Device

A thing made for a particular purpose; an invention or contrivance, especially a mechanical or electrical one.
[NIEM] DeviceType

Direct Supertypes

Item, Resource

1.24.5 Class Item

An inanimate material object as distinct from a living sentient being.
[NIEM] ItemType
[DOLCE] Non-agentive Physical Object

Direct Supertypes

Physical Entity

Attributes

height : Length
[NIEM]ItemHeightMeasure: A measurement of the height of an item.
A measurement in the vertical plane. For a person, from head to toe.

length : Length
[NIEM] ItemLengthMeasure: A measurement of the length of an item.
A longitudinal measurement - from end to end. Usually greater than width.

width : Length
[NIEM] ItemWidthMeasure: A measurement of the width of an item.
A horizontal measurement - from side to side.

1.24.6 Class Managed Item Identifier <<Value>>

[NIEM] An identification inscribed on or attached to a part, collection of parts, or complete unit by the manufacturer.
Syn. ItemSerialIdentification.
[FIBO] ProductIdentifier: an identifier for a product

Direct Supertypes

Unique Identifier

1.24.7 Class Physical Entity

A thing that exists in space and time including people, places, and things.

[DOLCE] Object

[IDEAS] Individual: A Thing that has spatio-temporal extent.

Note1 - this may be some that existed in the past, exists now, or may exist in some future possible world.

Note2 - the Individual may be scattered - i.e. it is the fusion of several disconnect parts.

Direct Supertypes

Spatial Entity

Attributes

weight : Mass

The current weight (as mass) of a physical thing.

1.24.8 Class Physical Feature

Physical features are spacial entities which are generically constantly dependent on physical objects (their hosts). Typical examples of features are “parasitic entities” such as holes, boundaries, surfaces, or stains. Physical features do not have mass independent of their host.

[DOLCE] Feature

Direct Supertypes

Spatial Entity

1.24.9 Class Physical Tool <<Role>>

An physical item intended to be used to perform some function.

Direct Supertypes

Item, Tool

1.24.10 Class Spatial Entity

A thing that exists in space: The union of locations and physical entities.

[DOLCE] Physical Endurant

Figure 79. Spacial Entity Detail

Direct Supertypes

Actual Entity

1.24.11 Class Telecommunication Device

A device for human to human communication over a distance by cable, telegraph, telephone, computer networks, or broadcasting.
[NIEM] TelecommunicationsDeviceType

Direct Supertypes

Communicating Device

Attributes

device address : Electronic Contact
An code or number used to communicate with or through a telecommunications device.

1.24.111 Enumeration Sex Kind

Kinds of sex. Eg. male/female.
package Concept Library::Physical Entities
public enum Sex Kind
{Male, Female}

Literals

- Male

A male person, plant, or animal. One able to fertilize a female with gametes.

- Female

A female person, plant, or animal. Of or denoting the sex that can bear offspring or produce eggs, distinguished biologically by the production of gametes (ova) that can be fertilized by male gametes:

Figure 80. Physical Entities

1.25 Concept Library::Places

This package defines concepts related to places. Places are buildings or localities used or intended for a purpose.

1.25.1 Diagram: Place

Figure 81. Place

1.25.2 Class Facility <<Role>>

[NIEM] FacilityType: A building, place, or structure that provides a particular service.

Direct Supertypes

Place

1.25.3 Association Class Operating Location <<Relationship>>

Place where an actor performs activities.

Figure 82. Operating Location

Direct Supertypes

Ability

Association Ends

performs at : Place [*]

Places where an actor perform activities.

utilized by : Actor [*]

Actors who utilizes a place to perform activities.

1.25.4 Class Place <<Role>>

A building or locality used or intended for a specific purpose such as a house or factory.
[FIBO] Facility: something that is built, contrived, established, or installed to serve a particular purpose, or make some course of action or operation easi-er, or provide some capability or service

Direct Supertypes

Contactable, Physical Location, Resource

1.25.5 Association Class Place of Occurrance <<Relationship>>

Relationship describing where something happens.

Figure 83. Place of Occurrance

Direct Supertypes

Impact

Association Ends

situated at : Place [0..*]

Place where a situation or event is located or happens.

location of : Situation [0..*]

Situations (Events, incidents, static arrangements, etc.) that happen at the subject place.

1.25.6 Class Residence <<Role>>

A place where people live/reside.

Direct Supertypes

Place

1.26 Concept Library::Policies

This package defines concepts related to policies. Policies deal with conditions asserted on one entity by another. This includes requirements and laws.

1.26.1 Diagram: Policy

Figure 84. Policy

1.26.2 Association Class Assertion of Policy <<Relationship>>

The assertion of a policy by an authority.

Figure 85. Assertion of Policy

Direct Supertypes

Assertion, Objective of Stakeholder, Subject to Authority

Association Ends

asserts policy : Policy [0..*]

A policy asserted by an authority whom states with authority that it must be followed.

asserted by : Authority [1..*]

The authority that asserts a policy, states with authority that it must be followed

1.26.3 Class Policy

A policy is a thing that is compulsory; a necessary condition.
A statement that identifies a necessary attribute, capability, characteristic, or quality of a system for it to have value and utility to a customer, organization, internal user, or other stakeholder. The constrained parties are identified as the <constrains> responsible performer(s).

A policy is a means in that it fulfills a broader objective. A policy is an objective in that performers seek to comply with the objective. A policy is a state in that it is a situation that exists for a finite period of time.

Policies include requirements.

[BMM] Business policy: directive that is concerned with directly controlling, influencing, or regulating the actions of an enterprise and the people in it and that is not directly enforceable

Direct Supertypes

Controlled Entity, Means, Objective

1.27 Concept Library::Predictions

Predictions are acts where an actor predicts that some possible situation will occur.

1.27.1 Diagram: Prediction

Figure 86. Prediction

1.27.2 Association Class Prediction <<Relationship>>

A prediction is a forecast that potential situations will happen.

Figure 87. Prediction

Direct Supertypes

Assertion

Association Ends

predicts : Potential Situation [1..*]

The situation that is postulated by a prediction.

predicted by : Predictor [0..*]

Predictor is the role of the actor who made a prediction.

Attributes

likelihood : Probability Metric

Metric representing the possibility that the containing element represents reality.

1.27.3 Class Predictor <<Role>>

The role of an actor making predictions.

Direct Supertypes

Social Agent

1.28 Concept Library::Processes

A package representing concepts about processes. Processes are templates for (descriptions of) a related sets of Events (activities, events, etc.). Processes may be natural, organizational or carried out by an actor. Processes carried out by an actor are plans.
Processes may require resources - noting that resource can be a role of any entity.
Processes are essentially patterns of Events.
Scenarios are typically less formal processes and describe how a series of Events may play out.

1.28.1 Diagram: Process

Figure 88. Process

1.28.2 Class Composite Process <<Role>>

Direct Supertypes

Occurrence Pattern, Temporal Composite

1.28.3 Association Class Invoke Process <<Relationship>>

The activity of initiating the performance of a process.
The process instance will be classified by the process.

Figure 89. Invoke Process

Direct Supertypes

Cause and Effect, Create, Process Action

Association Ends

creates process instance : Actual Occurrence [*]

An actual Event created as the instantiation of a process definition.

enactement of : Occurrence Pattern [0..1]

Process enacted by an Invoke Process

1.28.4 Class Modus Operandi

A particular way or method an actor typically does something, especially one that is characteristic or well-established. It may or may not have a formal definition of the process.
In threat terms, a particular tactic, technique or procedure for achieving a result.
Syn. TTP [STIX]

Direct Supertypes

Activity, Scenario

1.28.5 Class Occurrence Pattern

A process pattern is a template and definition for a family of Events (i.e. actions, events) that results in an outcome. A process may be natural or caused by the activities of actors, in which case it is a plan.
A process may contain other entities, sub-processes and situations to define characteristics and sub-processes of the

process. The sub-processes may or may not be known, sub-processes are defined using "Temporal Part".
[ISO 14971:2007] set of interrelated or interacting activities which transforms inputs into outputs

Direct Supertypes

Occurrence, Pattern

1.28.6 Class Plan

A plan is a design for a process that supports a stakeholders objectives. As a process definition a plan is a pattern for a series of activities as well as the resources required to meet objectives.
Scenario's are observed where as plans are designed.
[BMM] Course of Action: A Course of Action is an approach or plan for configuring some aspect of the enterprise involving things, processes, locations, people, timing, or motivation undertaken to achieve Desired Results. In other words, a Course of Action channels efforts towards Desired Results. To help ensure success in this regard, Courses of Action are governed by Directives.

Direct Supertypes

Means, Occurrence Pattern

1.28.7 Class Process Action

An action impacting a potential or realized process.

Direct Supertypes

Capability

1.28.8 Association Class Process Decomposition <<Relationship>>

Relationship describing the decomposition of a process.

Figure 90. Process Decomposition

Direct Supertypes

Temporal Part, Usage

Association Ends

has subprocess : Occurrence Pattern [0..*]

Process occurring within the scope of and in support a composite process.

used by process : Composite Process [0..*]

Composite processes which utilize the subject property as a component part.

1.28.9 Association Class When <<Relationship>>

A "When" rule defines an atomic process where by a <trigger> conditionally causes the <initiates> process to be invoked when the <trigger> process is matched under conditions(s) of the context - a proactive cause and effect. This results in an invocation of the <initiates> process.
E.g. when <trigger> do <initiates>
Also known as a "Course of action" or "ECA Rule" .

[PRR] ProductionRule: A ProductionRule is a statement of programming logic that specifies the execution of one or more actions in the case that its conditions are satisfied.

Figure 91. Course of Action Rule

Direct Supertypes

Cause and Effect, Conditional Rule, Occurrence Pattern

Association Ends

initiates : Occurrence Pattern [*]

Processes that will occur if the <trigger> process occurs. Consequent.

triggered by : Occurrence Pattern [*]

Processes that may cause the <initiates> process to occur. Antecedent.

1.29 Concept Library::Processes::Capabilities

Actions that impact various kinds of entities in specific ways. Such actions can be the subject of or part of processes, permissions, capabilities, or objectives.

1.29.1 Diagram: Capabilities

Figure 92. Capabilities

1.29.2 Class Capability <<Role>>

A particular ability or capacity that an actor may possess or exchange to achieve a specific purpose or outcome. Where a specific actor possesses the capability, the actor has an ability.

Adapted from: Ulrich Homann, "A Business-Oriented Foundation for Service Orientation", 2006.

Figure 93. Event Effecting Entity

Direct Supertypes

Activity

1.29.3 Class Create

The creation of something

Direct Supertypes

Capability

1.29.4 Class Damage

An action that causes an entity to no longer completely fulfill its purpose.

Direct Supertypes

Capability

1.29.5 Class Destroy

The destruction or deletion of something.

Direct Supertypes

Capability

1.29.6 Class Disrupt Process

An action intended to cause a process to not achieve its desired affect.

Direct Supertypes

Damage, Process Action

1.29.7 Class Entry Action

The action of entering through a boundary.

Direct Supertypes

Capability

1.29.8 Class Exit Action

An action of exiting through a boundary.

Direct Supertypes

Capability

1.29.9 Class Pause Process

An action that pauses a process instance such that it can be restarted.

Direct Supertypes

Process Action

1.29.10 Class Possible Actions

All possible effects to an entity.

Direct Supertypes

Capability

1.29.11 Class Stop Process

An action to terminate a process.

Direct Supertypes

Process Action

1.30 Concept Library::Processes::Composite Conditions

Composite conditions provide for "and"/"or" evaluation of causality between situations.

1.30.1 Diagram: Composite Condition

Figure 94. Composite Condition

1.30.2 Class AND Condition

A situation that is true (happening) only when all "enabled by" are true - AND

Direct Supertypes

Composite Condition

1.30.3 Class Composite Condition

A composite condition is a state that is inferred to be true or false based on the set of "affected by" (input) situations and the logic of the specific composite event subtype and the condition (if any).
The composite condition can then be used to trigger a set of "effects" (output) situations.
Combinations of Events, states. and composite conditions be combined with composite conditions to represent fault, flow or dependency graphs.
Note that cause & effect, negation and enablement are all subtypes of effect and can be combined using composite conditions.

Direct Supertypes

State

Attributes

negate : Boolean
Negates the logic of a complex event - NOT <condition>

1.30.4 Class OR Condition

A composite condition that is true (occurring) when any "affected by" is true - OR

Direct Supertypes

Composite Condition

1.30.5 Class XOR Condition

A state that is True only when exactly one of its "affected by" is true - XOR

Direct Supertypes

Composite Condition

1.31 Concept Library::Processes::Scenarios

Scenarios describe sequences and/or dependent situation patterns that include fault trees, attack/defense trees and other scenarios impacting stakeholders.

The attack-defense tree model is based on [Schweitzer 2013]
The attack-defense tree formalism can informally be described as follows. For a security scenario, the main goal of the attacker is depicted by the root node of a tree, an attack goal. Two types of refinement relations help to detail the model. Disjunctive (or) and constructive (and). Both types of refinement are recursively applied. Defense goals lower the probability of an attackers success, attack goals increase the probability.

1.31.1 Diagram: Scenarios

Figure 95. Scenarios

A performance goal that is an objective to disrupt some resource.

1.31.2 Class Goal State

A goal state is a step in a scenario that directly impacts resources of a stakeholder, thereby impacting the goals of the same or a different stakeholder. Goal states may be positive (e.g. making more money) or negative (e.g. an attack target). A goal state is usually but not necessarily the end state of a scenario.

Direct Supertypes

Consequence, Scenario Step

1.31.3 Class Scenario

A scenario is a process pattern that provides template for a set of steps (may be but are not always activities) and resource that formally or informally depict how things may happen based on observations of similar occurrences. Scenarios are intended to be descriptive, not prescriptive.
Scenarios are described by at least one scenarios steps, one of which must provide a goal state.
Scenarios include attack/defense and fault trees.

Direct Supertypes

Occurrence Pattern

1.31.4 Class Scenario Step

A scenario step is a node in one or more scenarios that ultimately result in consequences.

Direct Supertypes

Means, Situation

Attributes

likelihood : Probability Metric
Metric representing the probability that the scenario step will happen.

1.31.5 Association Class Step in Scenario <<Relationship>>

Definition of a particular step as comprising a particular scenario.

Figure 96. Step in Scenario

Direct Supertypes

Temporal Part

Association Ends

comprises step : Scenario Step [1..*]

A step in a scenario that is or leads to a goal state.

step in : Scenario [1..*]

Scenario in which is defined, in part, be a step.

1.32 Concept Library::Quantities and Units

This package defines quantities and units. Quantities are the basis for units and measurements.
Qualities of things are represented with respect to what that thing means, not how it is represented. This introduces multiple "quantity kinds" which derive from Value and Quantity. Quantiles are stereotyped as "Quantity Kind".
The representation of a value or quantity will typically use the "primitive types" that are found in I.T. systems such as "Integer", "Real" and "String".

.

1.32.1 Diagram: Quantities and units

Figure 97. Quantities and units

1.32.2 Class Confidence Metric <<Quantity Kind>>

Any metric of confidence that something is true or valid.

Direct Supertypes

Metric

1.32.3 Class Count <<Quantity Kind>>

The number of something used as a property or metric, e.g., 5 fish.

Direct Supertypes

Unit Value

1.32.4 Class Currency Benefit Metric <<Quantity Kind>>

A metric for benefit or harm expressed in terms of a currency, such as dollars or yen.

Direct Supertypes

Harm-Benefit Metric

1.32.5 Class Harm-Benefit Metric <<Quantity Kind>>

A metric to quantify benefit or harm.

Direct Supertypes

Metric

1.32.6 Class Metric <<Quantity Kind>>

A standard for measuring or evaluating something in a quantifiable way.
Typical representations of a metric may be a fraction from zero to 1 or a rating such as "high, medium, low". Not to be confused with the "Metric System".

Figure 98. Metric

Direct Supertypes

Unit Value

Attributes

value : Metric Value

The value of a quantity that, when multiplied by the unit defined in a subtype of quantity kind, specifies a measurement value such as 3 Meters.

1.32.7 Class Probability Metric <<Quantity Kind>>

A metric that represents the possibility that something uncertain will happen.

Direct Supertypes

Metric

1.32.8 Class Time Coordinate <<Quantity Kind>>

An identifier for a particular point in time, recognizing that any such point is an interval at a finer level of granularity. Specific time coordinate systems, such as ISO or Internet time, specialize Time Coordinate and relate it to a time scale.
[DTV] time point: concept that specializes the concept 'time interval' and that is a member of a time scale
[ISO11404] time: time is a family of datatypes whose values are points in time to various common resolutions: year, month, day, hour, minute, second, and fractions thereof.

Direct Supertypes

Coordinate, Unit Value

Attributes

- value : Duration

1.32.81 <<Value>>Enumeration PentaScale <<Value>>

An scale of 5 values the interpretation of which is context specific.

Direct Known Superclasses

Scale

package Concept Library::Quantities and Units
public enum PentaScale
{Very Low, Low, Moderate, High, Very High}

Literals

- Very Low
- Low
- Moderate
- High
- Very High

Figure 99. Quantities and units

1.32.82 <<Value>>Enumeration TriScale <<Value>>

A scale of 3 arbitrary levels.

Direct Known Superclasses

Scale

package Concept Library::Quantities and Units
public enum TriScale
{Low, Medium, High}

Literals

- Low
- Medium
- High

Figure 100. Quantities and units

1.33 Concept Library::Quantities and Units::Quantity Kinds

Quantity kinds are abstractions for the way we measure or quantify things, such as mass or length. Units provide specific ways to specify a quantity kind.

1.33.1 Diagram: Quantity Kinds

Figure 101. Quantity Kinds

1.33.2 Class Absorbed Dose (Radiation) <<Quantity Kind>>

The energy of ionizing radiation absorbed per unit mass by a body, often measured in rads.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.3 Class Acceleration <<Quantity Kind>>

The rate of change of velocity per unit of time.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.4 Class Amount of Substance <<Quantity Kind>>

The abstract unit of the amount of a substance which is the supertype of all amount units and also acts as its "quantity kind".

Amount of substance is a standards-defined quantity that measures the size of an ensemble of elementary entities, such as atoms, molecules, electrons, and other particles. It is sometimes referred to as chemical amount. The International System of Units (SI) defines the amount of substance to be proportional to the number of elementary entities present. The SI unit for amount of substance is the mole. It has the unit symbol mol.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.5 Class Angle <<Quantity Kind>>

The space (usually measured in radians or degrees) between two intersecting lines or surfaces at or close to the point where they meet.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.6 Class Area <<Quantity Kind>>

[QUDT] Area is a quantity expressing the two-dimensional size of a defined part of a surface, typically a region bounded by a closed curve.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.7 Class Color <<Quantity Kind>>

Color is the visual perceptual property corresponding in humans to the categories called red, blue, yellow, and others. Color derives from the spectrum of light (distribution of light power versus wavelength) interacting in the eye with the spectral sensitivities of the light receptors. Color categories and physical specifications of color are also associated with objects or materials based on their physical properties such as light absorption, reflection, or emission spectra. By defining a color space, colors can be identified numerically by their coordinates.

Direct Supertypes

Unit Value

1.33.8 Class Concentration <<Quantity Kind>>

The abstract concept of the amount, mass or volume of one substance in another without being specific as to how it is measured.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.9 Class Concentration (amount of substance) <<Quantity Kind>>

Concentration based on amount-of-substance.

Direct Supertypes

Concentration

1.33.10 Class Concentration (Mass) <<Quantity Kind>>

Concentration based on mass per unit of volume.

Direct Supertypes

Concentration

1.33.11 Class Concentration (Volume) <<Quantity Kind>>

Volume concentration is defined as the volume of a constituent divided by the volume of the mixture.

Direct Supertypes

Concentration

1.33.12 Class Currency <<Quantity Kind>>

Any form of money.
[FIBO] Currency: medium of exchange value, defined by reference to the geographical location of the authorities responsible for it

Direct Supertypes

Unit Value

1.33.13 Class Dose Equivalent (Radiation) <<Quantity Kind>>

A measure of the biological damage to living tissue as a result of radiation exposure. Also known as the "biological dose," the dose equivalent is calculated as the product of absorbed dose in tissue multiplied by a quality factor and then sometimes multiplied by other necessary modifying factors at the location of interest. The dose equivalent is expressed numerically in rems or sieverts (Sv) (see 10 CFR 20.1003). For additional information, see Doses in Our Daily Lives and Measuring Radiation. [NRC]
For practical purposes, 1 R (exposure) = 1 rad (absorbed dose) = 1 rem or 1000 mrem (dose equivalent).

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.14 Class Duration <<Quantity Kind>>

The abstract quantity kind of time which is the supertype of all time duration units.
Time is a measure that allows events to be ordered from the past through the present into the future, and also the measure of durations of events and the intervals between them. Durations are quantities of time, not points or intervals of time.

[DTV] base quantity of the International System of Quantities, used for measuring time intervals.

[IDEAS] Time: A MeasureInstance whose members are individuals' that have a particular temporal dimension of the same length.

[FIBO] Duration: An amount of time.

[UML] Duration

[OWL] xsd:duration

Direct Supertypes

Physical Quantity

1.33.15 Class Electric Current <<Quantity Kind>>

The abstract quantity kind of electric current which is the supertype of all current units.
[QUDT]Electric Current is the flow (movement) of electric charge. The amount of electric current through some surface, e.g., a section through a copper conductor, is defined as the amount of electric charge flowing through that surface over time. Current is a scalar-valued quantity.
The SI unit for measuring an electric current is the ampere, which is the flow of electric charge across a surface at the rate of one coulomb per second.

[IDEAS] ElectricCurrent: A MeasureInstance whose members are individuals' that all have the same electric current flowing through them

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.16 Class Electric Potential <<Quantity Kind>>

[QUDT] Electric Potential is a scalar valued quantity associated with an electric field.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.17 Class Energy <<Quantity Kind>>

The measure of energy- the ability to perform work (such as moving a mass).

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.18 Class Force <<Quantity Kind>>

(Physical) force is an influence that causes mass to accelerate. It may be experienced as a lift, a push, or a pull.
Force is defined by Newton's Second Law as $F = m \cdot a$, where F is force, m is mass and a is acceleration. Net force is mathematically equal to the time rate of change of the momentum of the body on which it acts. Since momentum is a vector quantity (has both a magnitude and direction).

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.19 Class Frequency <<Quantity Kind>>

Repetitions per unit of time. e.g., Hertz.

[IDEAS] Frequency: A MeasureInstance whose instances are individuals' that all oscillate at the same frequency

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.20 Class Length <<Quantity Kind>>

The abstract unit of distance (or length) which is the supertype of all length units and also acts as its "quantity kind".

In the International System of Quantities, length is any quantity with dimension distance. In other contexts "length" is the measured dimension of an object.

[IDEAS] Length: A MeasureInstance whose instances are individuals' that all have the same length

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.21 Class Luminosity <<Quantity Kind>>

Luminosity (or luminous intensity) is a measure of the wavelength-weighted power emitted by a light source in a particular direction per unit solid angle, based on the luminosity function, a standardized model of the sensitivity of the human eye. The SI unit of luminous intensity is the candela (cd), an SI base unit.

[IDEAS] LuminousIntensity: A MeasureInstance whose members are individuals' that all have the same luminous intensity

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.22 Class Mass <<Quantity Kind>>

The abstract unit of Mass which is the supertype of all mass units and also acts as its "quantity kind".
The mass of a body is a measure of its inertial property or how much matter it contains. The weight of a body is a measure of the force exerted on it by gravity or the force needed to support it. Gravity on earth gives a body a downward acceleration of about 9.8 m/s2.The SI unit of mass is the kilogram (kg).

[IDEAS] Mass: A MeasureInstance whose members are individuals' that all have the same mass.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.23 Class Mass Density <<Quantity Kind>>

The density, or more precisely, the volumetric mass density, of a substance is its mass per unit volume. The symbol most often used for density is ρ (the lower case Greek letter rho). Mathematically, density is defined as mass divided by volume.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.24 Class Physical Quantity <<Quantity Kind>>

A measurable property of a physical object.

Direct Supertypes

Unit Value

1.33.25 Class Power <<Quantity Kind>>

(Physical) power is the rate at which work is performed or energy is transmitted, or the amount of energy required or expended for a given unit of time. As a rate of change of work done or the energy of a subsystem, power is: P = W/t where P is power W is work t is time.

Direct Supertypes

Physical Quantity

1.33.26 Class Pressure <<Quantity Kind>>

A quantity kind representing the continuous physical force exerted on or against an object by something in contact with it.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.27 Class Radiation Exposure <<Quantity Kind>>

A measure of exposure to radiation.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.28 Class Radioactivity <<Quantity Kind>>

Radioactivity is a quantity kind that refers to the amount of ionizing radiation released by a material. Whether it emits alpha or beta particles, gamma rays, x-rays, or neutrons, a quantity of radioactive material is expressed in terms of its radioactivity (or simply its activity), which represents how many atoms in the material decay in a given time period. The units of measure for radioactivity are the curie (Ci) and Becquerel (Bq).

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.29 Class Speed <<Quantity Kind>>

A Quantity kind representing distance per unit of time.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.30 Class Temperature <<Quantity Kind>>

The abstract quantity kind of Thermodynamic temperature which is the supertype of all temperature units and also acts as its "quantity kind".
Thermodynamic temperature is the absolute measure of temperature and it is one of the principal parameters of thermodynamics.
Thermodynamic temperature is defined by the third law of thermodynamics in which the theoretically lowest temperature is the null or zero point.

[IDEAS] ThermodynamicTemperature:

Direct Supertypes

Physical Quantity, Scalar Quantity

1.33.31 Class Volume <<Quantity Kind>>

A quantity kind for the amount of space that a substance or object occupies.

Direct Supertypes

Physical Quantity, Scalar Quantity

1.34 Concept Library::Quantities and Units::Units

A package of common SI and U.S. Units. Note: All measures in concrete models should be bound to the expected units, even if units are implicit in the data structure.
This package is non-normative in the threat and risk specification. It is supplied to assist users in defining the units used in data structures.

1.34.1 Diagram: Common Units 1

Figure 102. Common Units 1

An amount of time.

1.34.2 Diagram: Common Units 2

Figure 103. Common Units 2

1.34.3 Class Acre <<Unit Value>>

1 acre = 43 560 square feet.

Direct Supertypes

Area

1.34.4 Class Ampere <<Base Unit Value>>

[NIST-SI] The ampere is that constant current which, if maintained in two straight parallel conductors of infinite length, of negligible circular cross-section, and placed 1 meter apart in a vacuum, would produce between these conductors a force equal to 2 x 10-7 newton per meter of length.

[IDEAS] electricCurrentInAmperes: A measureNamedNumericallyBy that names an ElectricCurrent with its ValueInAmperes

Direct Supertypes

Electric Current

1.34.5 Class Becquerel (Bq) <<Unit Value>>

The SI unit of radioactivity, corresponding to one disintegration per second.

Direct Supertypes

Radioactivity

1.34.6 Class Candela <<Base Unit Value>>

The candela is the luminous intensity, in a given direction, of a source that emits monochromatic radiation of frequency 540 x 1012 hertz and that has a radiant intensity in that direction of 1/683 watt per steradian.[NIST-SI]

[IDEAS] luminousIntensityInCandela: A measureNamedNumericallyBy that names a LuminousIntensity with its ValueInCandela

Direct Supertypes

Luminosity

1.34.7 Class Celsius <<Unit Value>>

Centigrade. The temperature scale (Celsius scale) in which 0° represents the ice point and 100° the steam point of water.

Direct Supertypes

Temperature

1.34.8 Class Compound Duration Value <<Unit Value>>

A value for a duration derived from the sum of primitive duration values. e.g. 2 days and 3 hours or 2016 years, 12 months and three days.

Direct Supertypes

Duration

Attributes

- value : Compound Duration

1.34.9 Class Concentration Percent <<Base Unit Value>>

The volume of a constituent divided by the volume of the mixture.

Direct Supertypes

Concentration (Volume)

1.34.10 Class Coulomb/kilogram (C/kg). <<Unit Value>>

Unit of radiation exposure.

Direct Supertypes

Radiation Exposure

1.34.11 Class Cubic Feet <<Unit Value>>

A volume measured in feet.

Direct Supertypes

Volume

1.34.12 Class Cubic Inch <<Unit Value>>

A volume measured in inches.

Direct Supertypes

Volume

1.34.13 Class Cubic Meter <<Base Unit Value>>

A volume measured in meters.

Direct Supertypes

Volume

1.34.14 Class Cup (US) <<Unit Value>>

8 Fluid Ounces (US).

Direct Supertypes

Liquid Volume

1.34.15 Class Curie (Ci) <<Base Unit Value>>

The SI unit of measure for radioactivity is the curie (Ci) and Becquerel (Bq).

Direct Supertypes

Radioactivity

1.34.16 Class Day <<Unit Value>>

A unit of time equal to 24 hours.

[DTV] day: the precise time unit that is quantified by 86 400 seconds
[OWL] xsd:gDay

Direct Supertypes

Scalar Duration Value

1.34.17 Class Degrees <<Unit Value>>

A unit for an angle from 0-360.

Direct Supertypes

Angle

1.34.18 Class Fahrenheit <<Unit Value>>

The Fahrenheit scale in which 32° represents the ice point and 212° the steam point. of water Symbol: F.

Direct Supertypes

Temperature

1.34.19 Class Fluid Ounce (US) <<Unit Value>>

A unit of volume: 16 fluid ounces = 1 pint (pt)
= 28.875 cubic inches.

Direct Supertypes

Liquid Volume

1.34.20 Class Foot <<Unit Value>>

A foot (pl. feet; abbreviation: ft; symbol: ′, the prime symbol) is a unit of length in the imperial and US customary systems of measurement. Since 1959, both units have been defined by international agreement as equivalent to 0.3048 meters exactly. In both systems, the foot comprises 12 inches and three feet compose a yard.

Direct Supertypes

Length

1.34.21 Class Gallon (Imperial) <<Unit Value>>

A unit of volume.

Direct Supertypes

Liquid Volume

1.34.22 Class Gallon (US) <<Unit Value>>

A measure of the quantity of a substance. 1 gallon (gal) = 231 cubic inches.

Direct Supertypes

Liquid Volume

1.34.23 Class Gram <<Unit Value>>

The gram is a SI unit of mass.

Direct Supertypes

Mass

1.34.24 Class Gray (Gy) <<Base Unit Value>>

[NRC] One of the two units used to measure the amount of radiation absorbed by an object or person, known as the "absorbed dose," which reflects the amount of energy that radioactive sources (with any type of ionizing radiation) deposit in materials (e.g., water, tissue, air) through which they pass. One gray (Gy) is the international system of units (SI) equivalent of 100 rads, which is equal to an absorbed dose of 1 Joule/kilogram. An absorbed dose of 0.01 Gy means that 1 gram of material absorbed 100 ergs of energy (a small but measurable amount) as a result of exposure to radiation.

Direct Supertypes

Absorbed Dose (Radiation)

1.34.25 Class Hertz <<Base Unit Value>>

A unit of frequency. Cycles per second.

[IDEAS] frequencyInHertz: A measureNamedNumericallyBy that names a Frequency with its ValueInHertz

Direct Supertypes

Frequency

1.34.26 Class Horsepower <<Unit Value>>

Horsepower (hp) is a unit of measurement of power (the rate at which work is done). There are many different standards and types of horsepower. This model uses the 746 watt interpretation of horsepower.

Direct Supertypes

Power

1.34.27 Class Hour <<Unit Value>>

A unit of time: 60 Minutes.

[DTV] hour: the precise time unit that is quantified by '3600 seconds'

Direct Supertypes

Scalar Duration Value

1.34.28 Class Inch <<Unit Value>>

A unit of length.

Direct Supertypes

Length

1.34.29 Class Joule <<Base Unit Value>>

The joule (symbol J, also called newton meter, watt second, or coulomb volt) is the SI unit of energy and work.

Direct Supertypes

Energy

1.34.30 Class Kelvin <<Base Unit Value>>

[NIST-SI] The kelvin, unit of thermodynamic temperature, is the fraction 1/273.16 of the thermodynamic temperature of the triple point of water.

[IDEAS] temperatureInKelvin: A measureNamedNumericallyBy that names a ThermodynamicTemperature with its ValueInKelvin

Direct Supertypes

Temperature

1.34.31 Class Kg per cubic meter <<Base Unit Value>>

The SI unit for density.

Direct Supertypes

Mass Density

1.34.32 Class Kilogram <<Base Unit Value>>

[NIST-SI] The kilogram is the unit of mass; it is equal to the mass of the international prototype of the kilogram. [NIST-SI]

[IDEAS] massInKilograms: A measureNamedNumericallyBy that names a Mass with its ValueInKilograms

Direct Supertypes

Mass

1.34.33 Class Kilogram per cubic meter <<Base Unit Value>>

The SI Unit of density.

Direct Supertypes

Concentration (Mass)

1.34.34 Class Kilometer <<Unit Value>>

A unit of length, the SI measure of distances equal to 1000 meters, and equivalent to 3280.8 feet or 0.621 mile. Symbol: km.

Direct Supertypes

Length

1.34.35 Class Kilometer per Hour <<Base Unit Value>>

The SI unit of speed

Direct Supertypes

Speed

1.34.36 Class Kilowatt hour <<Unit Value>>

The watt-hour (symbolized Wh) is a unit of energy equivalent to one watt (1 W) of power expended for one hour (1 h) of time.

Direct Supertypes

Energy

1.34.37 Class Liquid Volume <<Quantity Kind>>

Volume of a liquid.

Direct Supertypes

Volume

1.34.38 Class Meter <<Base Unit Value>>

The meter is the length of the path traveled by light in vacuum during a time interval of 1/299 792 458 of a second.[NIST-SI]

The meter, (SI unit symbol: m), is the fundamental unit of length in the International System of Units (SI).

[IDEAS] lengthInMeters: A measureNamedNumericallyBy that names a Mass with its ValueInKilograms

Direct Supertypes

Length

1.34.39 Class Meter per second squared <<Base Unit Value>>

The SI Unit of acceleration.

Direct Supertypes

Acceleration

1.34.40 Class Mile <<Unit Value>>

The mile is an English unit of length standardized as exactly 1.609344 kilometers.

Direct Supertypes

Length

1.34.41 Class Miles per Hour <<Unit Value>>

U.S. unit of speed.

Direct Supertypes

Speed

1.34.42 Class Millimeter <<Unit Value>>

A unit of length equal to one thousandth of a meter and equivalent to 0.03937 inch.
Abbreviation: mm.

Direct Supertypes

Length

1.34.43 Class Millisecond <<Unit Value>>

A unit of time: 1/1000th of a second.

[DTV] millisecond

Direct Supertypes

Scalar Duration Value

1.34.44 Class Minute <<Unit Value>>

A unit of time: 60 seconds.

[DTV] minute : the precise time unit that is quantified by '60 seconds'

Direct Supertypes

Scalar Duration Value

1.34.45 Class Mole <<Base Unit Value>>

The mole is a unit of measurement used in chemistry to express amounts of a chemical substance, defined as the amount of any substance that contains as many elementary entities (e.g., atoms, molecules, ions, electrons) as there are atoms in 12 grams of pure carbon-12.

Direct Supertypes

Amount of Substance

1.34.46 Class Mole Per Cubic Meter <<Base Unit Value>>

The SI unit for amount-of-substance concentration.

Direct Supertypes

Concentration (amount of substance)

1.34.47 Class Month <<Unit Value>>

[DTV] month: the nominal time unit that is the duration of a time interval required for one rotation of the Moon in its orbit around the Earth, approximated to a number of days.
[OWL] xsd:gMonth

Direct Supertypes

Scalar Duration Value

1.34.48 Class Newton <<Base Unit Value>>

The SI unit of force. Equivalent to 100,000 dynes. A Newton is equal to the force that would give a mass of one kilogram an acceleration of one meter per second per second.

Direct Supertypes

Force

1.34.49 Class Ounce-Mass (US) <<Unit Value>>

U.S. Unit of Ounce representing Mass.

Direct Supertypes

Mass

1.34.50 Class Pascal <<Base Unit Value>>

The SI unit of pressure, equal to one newton per square meter (approximately 0.000145 pounds per square inch, or 9.9 × 10-6 atmospheres).

Direct Supertypes

Pressure

1.34.51 Class Pint (US) <<Unit Value>>

Unit of liquid volume: 2 pints = 1 quart (qt) = 57.75 cubic inches.

Direct Supertypes

Liquid Volume

1.34.52 Class Pound-Force <<Unit Value>>

Pound-force is equal to the gravitational force exerted on a mass of one avoirdupois pound on the surface of Earth. Standard gravity is not constant but usually taken to be 9.80665 m/s2 (about 32.174 049 ft/s2) in the context of the surface of the earth.

Direct Supertypes

Force

1.34.53 Class Pound-Mass (Imperial) <<Unit Value>>

A unit of mass that is exactly 453.59237 grams.

Direct Supertypes

Mass

1.34.54 Class Pound-Mass (US lb) <<Unit Value>>

The pound avoirdupois, which forms the basis of the U.S. customary system of mass, is defined as exactly 453.59237 grams.
The avoirdupois pound is legally defined as a measure of mass, but the name pound is also applied to measures of force.
See also: http://www.nist.gov/pml/wmd/metric/upload/frn-59-5442-1959.pdf

Direct Supertypes

Mass

1.34.55 Class PSI <<Unit Value>>

Unit of pounds per square inch.

Direct Supertypes

Pressure

1.34.56 Class Quart (US) <<Unit Value>>

Unit of liquid volume where 4 quarts = 1 gallon (gal) = 231 cubic inches [NIST-UNITS].

Direct Supertypes

Liquid Volume

1.34.57 Class Radians <<Base Unit Value>>

A unit of an angle where there are 2 PI radians in a circle.

Direct Supertypes

Angle

1.34.58 Class Radiation Absorbed Dose (rad) <<Unit Value>>

One of the two units used to measure the amount of radiation absorbed by an object or person, known as the "absorbed dose," which reflects the amount of energy that radioactive sources deposit in materials through which they pass. The radiation-absorbed dose (rad) is the amount of energy (from any type of ionizing radiation) deposited in any medium (e.g., water, tissue, air). An absorbed dose of 1 rad means that 1 gram of material absorbed 100 ergs of energy (a small but measurable amount) as a result of exposure to radiation. The related international system unit is the gray (Gy), where 1 Gy is equivalent to 100 rad. For additional information, see Doses in Our Daily Lives and Measuring Radiation. [NRC]

Direct Supertypes

Absorbed Dose (Radiation)

1.34.59 Class Roentgen (R) <<Base Unit Value>>

A unit of exposure to ionizing radiation. It is the amount of gamma or x-rays required to produce ions resulting in a charge of 0.000258 coulombs/kilogram of air under standard conditions. [NRC]

Direct Supertypes

Radiation Exposure

1.34.60 Class Roentgen Equivalent Man (REM) <<Unit Value>>

One of the two standard units used to measure the dose equivalent (or effective dose), which combines the amount of energy (from any type of ionizing radiation that is deposited in human tissue), along with the medical effects of the given type of radiation. For beta and gamma radiation, the dose equivalent is the same as the absorbed dose. By contrast, the

dose equivalent is larger than the absorbed dose for alpha and neutron radiation, because these types of radiation are more damaging to the human body. Thus, the dose equivalent (in rems) is equal to the absorbed dose (in rads) multiplied by the quality factor of the type of radiation [see Title 10, Section 20.1004, of the Code of Federal Regulations (10 CFR 20.1004), "Units of Radiation Dose"]. The related international system unit is the sievert (Sv), where 100 rem is equivalent to 1 Sv. [NRC]

Direct Supertypes

Dose Equivalent (Radiation)

1.34.61 Class Scalar Duration Value <<Unit Value>>

Number and time unit together giving magnitude of a duration.

[DTV]
Definition:if the atomic duration value is a precise atomic duration value, then the time unit is the reference duration to which the ratio of the duration quantified by the atomic duration value is taken
Definition:if the atomic duration value is a nominal atomic duration value, then the time unit is the reference duration to which the ratio of exactly one element of the duration value set specified by the atomic duration value is taken
Example:“45 minutes” has the time unit ‘minute’

Figure 104. Scalar Duration Value

Direct Supertypes

Duration, Scalar Quantity

1.34.62 Class Second <<Base Unit Value>>

[NIST-SI] The second (symbol: s) is the base unit of time in the International System of Units (SI) and is also a unit of time in other systems of measurement (abbreviated s or sec); it is the second division of the hour by sixty, the first division by 60 being the minute.

[DTV] second: The second is the duration of 9 192 631 770 periods of the radiation corresponding to the transition between the two hyperfine levels of the ground state of the cesium 133 atom.

[IDEAS] Second: A measureNamedNumericallyBy that names a Time with its ValueInSeconds

Direct Supertypes

Scalar Duration Value

1.34.63 Class Sievert (Sv), <<Base Unit Value>>

The international system (SI) unit for dose equivalent equal to 1 Joule/kilogram. 1 sievert = 100 rem. Named for physicist Rolf Sievert.

Direct Supertypes

Dose Equivalent (Radiation)

1.34.64 Class Square Feet <<Unit Value>>

Area measured in feet.

Direct Supertypes

Area

1.34.65 Class Square Meter <<Base Unit Value>>

Area measured in SI meters.

Direct Supertypes

Area

1.34.66 Class Volt <<Base Unit Value>>

The SI unit of electromotive force, the difference of potential that would drive one ampere of current against one ohm resistance.

Direct Supertypes

Electric Potential

1.34.67 Class Watt <<Base Unit Value>>

The SI unit of power is the joule per second (J/s).

Direct Supertypes

Power

1.34.68 Class Yard <<Unit Value>>

A Unit of length equal to 3 feet.[NIST-UNITS]

Direct Supertypes

Length

1.34.69 Class Year <<Unit Value>>

The period of 365 days (or 366 days in leap years) starting from the first of January, used for reckoning time in ordinary affairs.

[DTV] year: the nominal time unit that is the duration of a time interval required for one revolution of the Earth around the Sun, approximated to an integral number of days
[OWL] xsd:gYear

Direct Supertypes

Scalar Duration Value

1.35 Concept Library::Resources

This package represents concepts concerning resources. A resource is a role of any entity such that it supports or impacts in a process, impacts the objectives of stakeholders or is the basis of the capability of an actor.
As a role, “Resource” is intended to "mix in" with an entity type such as "Person" or "Process" such that the use of that entity may be understood.
Resources that are a Primary Asset are those that are the direct subject of a stakeholder's objectives.

1.35.1 Diagram: Resource

Figure 105. Resource

1.35.2 Diagram: Resource Actions

Figure 106. Resource Actions

1.35.3 Class Abuse Resource

An action to misuse a resource.

Direct Supertypes

Resource Actions

1.35.4 Class Capture Resource

Action to capture or gain control of some resource.

Direct Supertypes

Resource Actions

1.35.5 Class Damage Resource

An action that causes an resource to no longer completely fulfill its purpose..

Direct Supertypes

Damage, Resource Actions

1.35.6 Class Exceed Resource Capacity

An action to exceed the capacity of some resource.

Direct Supertypes

Resource Actions

1.35.7 Class Modify Resource

Action to modify a resource or set of resources.

Direct Supertypes

Resource Actions

1.35.8 Class Performer <<Role>>

A performer is an actor that is a resource to another entity as the performer of activities.

Direct Supertypes

Actor, Resource

1.35.9 Class Resource <<Role>>

A resource is a role of an entity required for or helpful to any operation, activity, process or capability - directly or indirectly. Sometimes called an "asset".

Direct Supertypes

Entity, Means

1.35.10 Class Resource Actions

An action impacting a potential or realized resource/asset.

Direct Supertypes

Capability

1.35.11 Association Class Resource Dependency <<Relationship>>

Relationship between resources where one resource depends on (or uses) another.
A more general concept than [UAF] MapsToCapability: An Abstraction relationship denoting that an Activity contributes to providing a Capability.

[DOLCE] (Subtype of) Dependence

Figure 107. Resource Dependency

Direct Supertypes

Impact

Association Ends

supports : Resource [*]

Resources the subject resource supports or enables.

depends on : Resource [*]

A resource that is required to support the operation of purpose of another resource.

1.35.12 Class Tool <<Role>>

The role of some inanimate thing used to facilitate a process or activity by an actor performing a process or activity.

Direct Supertypes

Resource

1.36 Concept Library::Response

1.36.1 Diagram: Response

Figure 108. Response

1.36.2 Association Class Initiation of Response <<Relationship>>

Figure 109. Initiation of Response

Direct Supertypes

Cause and Effect

Association Ends

has response : Response [0..*]

responds to : Situation [1..*]

1.36.3 Class Response <<Role>>

A response is an activity performed by an actor as a reaction to (caused by) another situation.

Direct Supertypes

Activity

1.37 Concept Library::Situations

Concepts relative to situations. A situation is a particular configuration of things and their relations including spatial, temporal, and logical connections between those things valid over a period of time. Situations form the basis of all complex, time dependent entities.

1.37.1 Diagram: Actual Situations

Figure 110. Actual Situations

1.37.2 Diagram: Situation Relationships

package Situations[Situation Relationships]
«Value»
Identifier
«Relationship»
Impact
attributes
likelihood : Probability Metric
impacts
impacted by
Entity
relates to
related from
«Relationship»
Temporally Related
affects
{subsets impacts}
«Role Of»
«Role»
Composite
Temporal Region
is after
Temporal Order
is before
start of
Start Time
starts at
0..1
finishes at
Finish Time
0..1
finish of
overlaps from
Overlaps in Time
overlaps to
duration of
Duration of Entity
«Characteristic»
has duration
1
Duration
Actual Entity
Assertion
holds within
asserts
Context
Proposition
«Relationship»
Effect
«Intersection»
Temporal Entity
Situation
affected by
«Relationship»
Enablement
enables
enabled by
«Relationship»
Involvement
involved in
involves
Actor
«Relationship»
Cause and Effect
caused by
casuses
negated by
0..*
negates
0..*
«Relationship»
Negation Effect
{complete, disjoint}
Event or State Constraint
State
Occurrence
Situation Kinds
«Intersection»
Actual Situation
has state
State of Entity
state of
Entity

Figure 111. Situation Relationships

This diagram shows the primary associations defined for situations as well as its super types: Temporal Entity and Identifiable Entity. The relationships shown here are those deemed defining for the concept of a situation, they are not all the relationships defined for these types.

1.37.3 Diagram: Situation Timeframes

Figure 112. Situation Timeframes

1.37.4 Diagram: Situations

Figure 113. Situations

1.37.5 Class Actual Situation <<Intersection>>

An actual situation is an individual situation that actually exists, happened in the past or may exist in some possible world, not a template or process definition. Such situations must exist for a time interval, however there are no constraints on such a time interval - from an instant to the life of the universe.

DTV: Occurrence: state of affairs that is a happening in the universe of discourse

Figure 114. Actual Situation

Direct Supertypes

Situation, Temporal Entity

1.37.6 Class Actual State <<Intersection>>

A condition that has, will or does exist.

Direct Supertypes

Actual Situation, State

1.37.7 Association Class Cause and Effect <<Relationship>>

The causality relation where the <causes> situation is <caused by> a situation.

[FIBO] cause / caused by

[ISO 1087] causal relation: associative relation (3.2.23) involving cause and its effect
NOTE A causal relation exists between the concepts (3.2.1) 'action' and 'reaction', 'nuclear explosion' and 'fall-out'.

Figure 115. Cause and Effect

Direct Supertypes

Enablement

Association Ends

caused by : Situation [*]

One of situations that causes the subject situation.

casuses : Situation [*]

A situation caused by another.

1.37.8 Class Current Situation

A situation that is actually occurring at the moment. "the moment" is contextual and interpreted within the context of the model.

Direct Supertypes

Actual Situation, Situation

1.37.9 Association Class Effect <<Relationship>>

Any impact on or alteration of an entity by a situation - an effect of the situation.

Figure 116. Effect

Direct Supertypes

Impact

Association Ends

affects : Entity [*]

Entities affected by a action

affected by : Situation [*]

Actions that can cause some change in a related entity.

1.37.10 Association Class Enablement <<Relationship>>

A situation that enables (is a condition for), another.

Figure 117. Enablement

Direct Supertypes

Effect, Impact

Association Ends

enables : Situation [*]

A situation that is enabled by another.

enabled by : Situation [*]

A situation that enables another.

1.37.11 Association Class Involvement <<Relationship>>

The relationship between an actor and situations they are involved in.
[DOLCE] Participation

Figure 118. Involvement

Direct Supertypes

Temporally Related

Association Ends

involved in : Situation [*]

Situations in which an actor has any kind of involvement.

involves : Actor [*]

An actor involved in a situation in any way.

1.37.12 Association Class Negation Effect <<Relationship>>

The negative causality relationship - <negated by> prevents or terminates the <negates> situation.

Figure 119. Negation

Direct Supertypes

Effect, Negation

Association Ends

negated by : Situation [0..*]

A situation that prevents or terminates another.

negates : Situation [0..*]

A situation that is prevented or terminated by another situation.

1.37.13 Class Past Situation

A situation that has actually occurred in the past (recognizing that all such statements are subject to confidence).

Direct Supertypes

Actual Situation, Situation

1.37.14 Class Potential Situation

A situation that has not yet happened but has a potential to happen.

DTV: Situation Kind

Direct Supertypes

Pattern, Situation

Attributes

likelihood : Probability Metric

Metric representing the possibility that the containing element represents reality.

1.37.15 Class Situation

A situation is an identifiable entity composed of an arrangement of entities and the relations between them over a time interval. Situations are propositions and may be asserted as true or false in some context. Situations may change over time, unless otherwise constrained. As an identifiable entity, situations may participate in relationships, thus situations are "first class" elements in SMIF.

[SBVR] "State of affairs"
[SOWA1999] Nexus

Direct Supertypes

Context, Lexical Context, Proposition

1.37.16 Class Situation Type

A situation type defines a kind of identifiable arrangement of individuals, assertions and the relations between them over a timespan. As an identifiable entity, situations may participate in other situations and relationships by being bound to properties of those situations or relationships with bindings, thus situations are “first class” entities in a SMIF model.
The roles or behaviors things (any entity or value) may play in a situation are identified as properties of the situation type.
Entity types and roles may also be situation types.
Syn. Type of a state of affairs.
A situation type may have properties such that instances, may bind things to structures based on properties.
Things may be bound to a structure (i.e. play a role in the structure) via properties. Things bound to properties of a structure may change over time, unless otherwise constrained.

[DTV] situation kind: state of affairs that may or may not happen in some possible world

Direct Supertypes

Entity Type

1.37.17 Class State

A state is a static situation - a particular configuration of entities that is static for a time period, including spatial and logical connections between those things {Snapshot of a Perdurant}
Note that states may be of any length, from an instant to infinity and beyond.

[DOLCE] State

Direct Supertypes

Situation

1.37.18 Association State of Entity

Relationship between a perdurant (something that exits over time) and a "state" of that entity as a snapshot in time.

Association Ends

state of : Entity [*]

The endurant entity for which the subject state is a snapshot.

has state : State [*]

A states (or snapshots) of an entity within its lifetime.

1.38 Concept Library::Social Agents

Actor relationships augment the concept of an actor with concepts of identifiers and associations between actors, including organization membership. See also the base actor class.

1.38.1 Diagram: Social Agent

Figure 120. Social Agent

Actor associations define the concept of an organization and provide a general framework for associations between actors with the use of an Actor Association.

1.38.2 Diagram: Social Agent Identifiers

Figure 121. Social Agent Identifiers

1.38.3 Association Class Assertion <<Relationship>>

The act of making a claim that something is true.

Direct Supertypes

Actual Occurrence, Temporally Related

Association Ends

asserts : Situation [0..*]

A situation asserted to be true by a social agent.

asserted by : Social Agent [0..*]

A social agent claiming the truth of a situation.

1.38.4 Class Associate <<Role>>

Direct Supertypes

Social Agent

1.38.5 Association Class Associated Actor <<Relationship>>

An associated actor relationship defines a connection between an actor and some other actor they are associated with in some way. Subtypes of actor association provide additional semantics about the association. As an association class, Actor Associations may have properties and other relationships. Actor associations will typicaly have a timeframe.

[NIEM] PersonOrganizationAssociationType (More specific concept)
[NIEM] PersonPersonAssociationType (More specific concept)

Figure 122. Associated Actor

Direct Supertypes

Temporally Related

Association Ends

has associate : Associate [*]

The actor associated with another.

associated with : Social Agent [1..*]

Another actor the subject actor is associated with.

1.38.6 Class License Identifier <<Value>>

[NIEM] An identification that references a license certification or registration of a person or organization for some purpose.

Direct Supertypes

Managed Social Agent Identifier

1.38.7 Class Local Identifier <<Value>>

An identification assigned at a local level (within an organization or community) to a person or organization.

Direct Supertypes

Managed Social Agent Identifier

1.38.8 Class Managed Social Agent Identifier <<Value>>

An identifier for a social agent where the identifier is managed by some authority.

Direct Supertypes

Managed Actor Identifier

1.38.9 Class Member <<Role>>

Direct Supertypes

Associate

1.38.10 Class Regional Identifier <<Value>>

[NIEM] An identification of a person based on a regional ID.

Direct Supertypes

Managed Social Agent Identifier

1.38.11 Class Social Agent

An actor that may have responsibilities - people and organizations. Actors in general may include automated entities and even, in some context, animals. Social agent excludes these other kinds of actors by including (at this time) only people and organizations.

What responsibilities a particular person or organization may have at any particular time is the subject of law and social constructs. A social agent is distinguished in that a person or organization may have such responsibilities in their lifetime.

[NIEM] EntityType

[DOLCE] Social Agent

Direct Supertypes

Actor

1.38.12 Class Tax Authority Identifier <<Value>>

An identifier assigned to a person or organization by a tax authority.

Direct Supertypes

Managed Social Agent Identifier

1.39 Concept Library::Systems

A system is a collection of parts and relationships among these parts that may be organized to accomplish some purpose. The term 'system' can refer to an information processing system but it is also applied more generally. Thus a system may include anything: a system of hardware, software, an enterprise, a federation of enterprises, a business process, some combination of parts of different systems, a federation of systems - each under separate control, a program in a computer, a system of programs, a single computer, a system of computers, a computer or system of computers embedded in some machine, etc.

One of the key strengths of modeling, and one that distinguishes it from implementation technologies like software source code, is that it is an excellent way to represent, understand, and specify systems. [OMG MDA Guide]

1.39.1 Diagram: System

Figure 123. System

1.39.2 Class Access Point

A point of entry into or out of a system such as a door, gate, port, or "interface" into an information system.

Direct Supertypes

Resource

1.39.3 Class Boundary

Something on the edge of a system that serves to contain or protect components of that system.
Some boundaries protect resources of the system/enterprise and may also be countermeasures.
e.g., Fences, firewalls, skin.

Direct Supertypes

Resource

1.39.4 Association Class Boundary of System <<Relationship>>

An "edge" of a system affording some level of protection or containment for the system.

Figure 124. Boundary of System

Direct Supertypes

Parthood

Association Ends

has boundary : Boundary [0..*]

Logical or physical border of a system (or enterprise) that may serve to define, contain or protect the system.

bounds : System [1..*]

System for which a boundary is an edge.

1.39.5 Association Class Opening in a Boundary <<Relationship>>

An access point that provides an opening for passage through a boundary. An opening has the potential to cause a point of entry and a vulnerability.

Figure 125. Opening in a Boundary

Direct Supertypes

Parthood

Association Ends

provides access through : Boundary [1..*]

Boundary through which an entry point provides access. e.g., the wall a door goes through.

has opening : Access Point [*]

A physical or virtual place where access to a system may be provided through a boundary.

1.39.6 Association Class Point Of Entry <<Relationship>>

Relationship between a system and a usable entry point into or out of that system. A point of entry has the potential to cause a vulnerability.

Figure 126. Point Of Entry

Direct Supertypes

Impact

Association Ends

enters into : System [*]

System into which an entry point provides access. E.G. A room a door enters into.

has portal : Access Point [*]

A point of possible entry into a system.

1.39.7 Class Subsystem <<Role>>

A role of a system as a part of another system as a resource to the owning system.

Direct Supertypes

Resource, System

1.39.8 Class System

[OMG MDA Guide] A system is a collection of parts and relationships among these parts that may be organized to accomplish some purpose.
[UAF] An integrated set of elements, subsystems, or assemblies that accomplish a defined objective. These elements include products (hardware, software, firmware), processes, people, information, techniques, facilities, services, and other support elements .
A system is a situation in that it has constituent parts working together for a finite period.
A system is a means in that it may achieve objectives for stakeholders.

Direct Supertypes

Actor, Composite, Means, Situation

1.40 Concept Library::Time & Temporal Entities

The Time package defines the essential concepts of time and the identification of time intervals.

These time concepts are based on the OMG Date Time Vocabulary [DTV] standard but subsets and simplifies DTV for use in defining, federating and exchanging time aspects of entities.

"Temporal Entity" is introduced as an abstraction to capture the common relationships between time elements. Within DTV these relationships are separate for each kind of time element. The relationships defined for Temporal Entity are grounded in DTV Time Interval as each temporal entity exists for a time interval.

Applications that need to reason about time are encouraged to utilize the full DTV semantics. DTV also contains text that more fully elaborates time concepts.

1.40.1 Diagram: Time

Figure 127. Time

1.40.2 Class Date and Time <<Quantity Kind>>

[FIBO] DateTimeStamp: A DateTimeStamp combines a Date, a time, and a time

Direct Supertypes

Date Coordinate, Time Coordinate

1.40.3 Class Date Coordinate <<Quantity Kind>>

[FIBO] Date: A Date identifies a calendar day on some calendar.
[NIEM] DateType

Direct Supertypes

Time Coordinate

1.40.4 Association Duration of Entity

[DTV] time interval [of temporal entity] has particular duration:the particular duration is the duration that is the amount of time in the time interval.

Each time interval [Temporal Entity] has a unique duration attribute that is a measure of its size, i.e., the amount of time the time interval occupies. This attribute is mathematically a function that maps time intervals into durations. This mapping function is sometimes called the “range” of a time interval, and some times called the “measure” of a time interval.

Association Ends

duration of : Temporal Region [*]

Temporal entity for which a duration is applicable.

has duration : Duration [1]

Difference between the start and end time. A non-zero positive value representing the amount of time a temporal entity exists.

[DTV] time interval [of temporal entity] has particular duration:the particular duration is the duration that is the amount of time in the time interval.

Each time interval [Temporal Entity] has a unique duration attribute that is a measure of its size, i.e., the amount of time the time interval occupies. This attribute is mathematically a function that maps time intervals into durations. This mapping function is sometimes called the “range” of a time interval, and some times called the “measure” of a time interval.

1.40.5 Association Entity Exists for Interval

Relationship defining the time interval in which an entity actually exists.

[DTV] occurrence occurs for occurrence interval: the occurrence occurs throughout the occurrence interval and the occurrence does not occur within some time interval2 that meets the occurrence interval and the occurrence does not occur within some time interval3 that is met by the
occurrence interval

[Alen 1983] X is equal to Y: X = Y where X is "interval of" and Y is "exists for". Note that this relates the temporal aspect of "interval of" to the timeframe "exists for".

Figure 128. Entity Exists for Interval

Direct Supertypes

Overlaps in Time

Association Ends

exists for : Time Interval [1]

Time interval where an entity may be considered "actual", that is existent in the domain of discourse.

interval of : Temporal Entity [*]

Entity existent for the full extent of a time interval.

1.40.6 Association Finish Time

The time something no longer exists (inclusive).

[DTV] time interval1 finishes time interval2
Synonymous Form:time interval2 is finished by time interval1
Definition:time interval1 is a proper part of time interval2 and there exists no time interval3 that is a proper part of time interval2 and that is after time interval1

[IDEAS] endBoundary: A temporalBoundary where the boundary is a end boundary of the whole.

[Alen 1983] X finishes Y: X f Y where X is "finishes at" and Y is "finish of"

Figure 129. Finish Time

Direct Supertypes

Overlaps in Time

Association Ends

finish of : Temporal Region [*]

Thing which no longer exists at a particular time.

finishes at : Temporal Region [0..1]

Time something no longer exists. (Inclusive)

1.40.7 Association Overlaps in Time

Some or all parts of the related temporal entities exist at the same time. Note that "to" and "from" may be arbitrary. By convention, the containing or prior temporal entity is "from".

[DTV] time interval1 properly overlaps time interval2

An [ISO 1087] temporal relation: sequential relation (3.2.24) involving events in time

[DOLCE] (subtype of) Temporal Quality

[Alen] X overlaps with Y: X o Y, where assignment of X and Y is arbitrary.

Figure 130. Overlaps in Time

Association Ends

overlaps from : Temporal Region [*]

An overlapping temporal component.

overlaps to : Temporal Region [*]

An overlapping temporal component.

1.40.8 Association Start Time

The time something starts to exist (inclusive).

[DTV] time interval1 starts time interval2
Synonymous Form:time interval2 is started by time interval1
Definition:time interval1 is a proper part of time interval2 and there exists no time interval3 that is a proper part of time interval2 and that is before time interval1.

[IDEAS] startBoundary: A temporalBoundary where the boundary is a start boundary of the whole.

[Alen 1983] X starts Y: X s Y where X is "starts at" and Y is "start of".

Figure 131. Start Time

Direct Supertypes

Overlaps in Time

Association Ends

start of : Temporal Region [*]

Thing which begins to exist at a particular time.

starts at : Temporal Region [0..1]

Time somethings begins to exist. (inclusive).

[FIBO] hasStartDate

1.40.9 Class Temporal Composite <<Role>>

Direct Supertypes

Composite, Temporal Region

1.40.10 Class Temporal Entity <<Intersection>>

An actual entity existing for a duration in time.

Direct Supertypes

Actual Entity, Temporal Region

1.40.11 Association Temporal Order

A relationship representing ordering of temporal entities in time where the <starts at> of <is after> is greater than or equal to the <finishes at> of <is before>. Related things do not overlap in time.

[DOLCE] (subtype of) Temporal Quality

[DTV] "time interval1 is properly before time interval2": the time interval1 is before the time interval2 and the time interval1 is before a time interval3 and the time interval3 is before the time interval2

[DTV] time interval1 finishes duration after time interval2: The end of one time interval is duration after the end of the other time interval.

[IDEAS] beforeAfter: A couple that asserts one individuals' temporal extent is completely before the temporal extent of another.

An [ISO 1087] temporal relation: sequential relation (3.2.24) involving events in time

[Alen1983] X takes place before Y: X<Y where X is "is before" and Y is "is after".

Figure 132. Temporal Order

Association Ends

is after : Temporal Region [*]

A temporal entity that starts after the <is before> entity ends.

is before : Temporal Region [*]

A temporal entity that ends after the <is after> entity starts.

1.40.12 Association Class Temporal Part <<Relationship>>

The time interval of <has temporal part> is within the time interval of <happens durring>.

[DTV] time interval1 is proper part of time interval2: the time interval1 is a proper part of the time interval2 and a time interval3 is a proper part of the time interval2 and a time interval4 is a proper part of the time interval2 and the time interval3 is before the time interval1 and the time interval1 is before the time interval4.

[IDAS] temporalWholePart: A wholePart that asserts the spatial extent of the (whole) individual is co-extensive with the spatial extent of the (part) individual for a particular period of time.

[Alen 1983] X during Y: Xd Y where X is "happens during" and Y is "has temporal part"

Figure 133. Temporal Part

Direct Supertypes

Overlaps in Time, Parthood

Association Ends

happens during : Temporal Composite [*]

A situation with overlapping duration (overlapping temporal extent).

has temporal part : Temporal Region [1..*]

Sub-durations of anything that happen - a temporal part.

1.40.13 Class Temporal Region

A temporal is anything that has a timespan. Temporal things may have temporal relationships with other temporal things.

Note that relationships defined for [DTV] Time Intervals may be specified for <temporal Entity> but are not specified in SMIF.

[SOWA1999] Continuant

Figure 134. Temporal Entity Detail

Direct Supertypes

Entity

1.40.14 Class Time Coordinate <<Quantity Kind>>

A designation of a particular time.

Direct Supertypes

Time Coordinate

1.40.15 Class Time Interval

A segment of time.

[DTV] "time interval" : segment of the time axis, a location in time.

Note:Every time interval has a beginning, an end, and a duration, even if not known. Every time interval is “finite”, a bounded segment of the Time Axis. The beginning or end of a
time interval may be defined by reference to events that occur for a time interval that is not known.

Note:Time intervals may be ‘indefinite’, meaning that their beginning is ‘primordiality’ or their end is ‘perpetuity’, or both (‘eternity’). This vocabulary assumes that indefinite
time intervals exist and have some duration, but their duration is unknown.

[IDEAS] PeriodOrInstant: An Individual whose spatial extent is infinite, but whose temporal extent is finite or zero.

[UML] TimeInterval
[NIEM] DateRangeType
[DOLCE] Temporal Region

Direct Supertypes

Context, Temporal Region

1.40.16 Class Time Point

A portion of time deemed atomic on a time scale. As all points in time may be further subdivided into a finer granularity of time, each point in time is also a time interval on some other scale.

The duration of a time point is the same as the granularity of the time scale of the time point.

[DTV] time point: concept that specializes the concept 'time interval' and that is a member of a time scale.

[IDEAS] CalendarPeriod: A Period that corresponds to a recognized date or time.

Direct Supertypes

Time Interval

1.40.17 Class Time Scale

A time scale is a way to reckon time as a series of consecutive time points identified by time coordinates. e.g. Time scale defined by the Gregorian calendar.

[DTV] time scale: regular sequence that each member of the regular sequence is a time point

Direct Supertypes

Coordinate System

1.40.18 Association Time Scale Granularity

[DTV] Time scale has granularity: The granularity of the time scale is the duration of the time points of the time scale.

Association Ends

has granularity : Duration [1]

[DTV] the smallest duration that can be distinguished with a given time scale

granularity of : Time Scale [*]

Duration of each time point on a time scale.

1.40.19 Association Time Scale of Time Point

Relationship defining the time scale on which a time point is defined. e.g. December 7th, 1944 is defined on a Gregorian Calendar time scale.

Association Ends

time point on : Time Scale [1]

Time scale used for defining a time point.

[DTV] time scale has time point:

has member time Point : Time Point [1..*]

Time point defined within a time scale

1.41 Concept Library::Time & Temporal Entities::ISO Time Scale

1.41.1 Diagram: ISO Time

Figure 135. ISO Time

1.41.2 Class Date Time Coordinate (ISO 8601) <<Unit Value>>

[UAF] A date and time specified in the ISO8601 date-time format including timezone designator (TZD): YYYY-MM-DDThh:mm:ssTZD.

Direct Supertypes

Date and Time

Attributes

value : Coded Numeric

A text string representing a date and time specified in the ISO8601 date-time format including timezone designator (TZD): YYYY-MM-DDThh:mm:ssTZD.

1.42 Concept Library::Time & Temporal Entities::XSD Time Scale

XSD Representations of date and time

1.42.1 Diagram: XSD Time Scale

Figure 136. XSD Time Scale

1.42.2 Class XSD Date <<Unit Value>>

An XSD representation of a date
[OWL] xsd:date.

Direct Supertypes

Date Coordinate

Attributes

value : Coded Numeric

A text string representing a date and time specified in the ISO8601 date-time format including timezone designator (TZD): YYYY-MM-DDThh:mm:ssTZD.

1.42.3 Class XSD Date Time <<Unit Value>>

An XSD representation of a date and time
[OWL] xsd:dateTime

Direct Supertypes

Date and Time

Attributes

value : Coded Numeric

A text string representing a date and time specified in the ISO8601 date-time format including timezone designator (TZD): YYYY-MM-DDThh:mm:ssTZD.

1.42.4 Class XSD Time <<Unit Value>>

An XSD representation of a time

[OWL] xsd:time

Direct Supertypes

Time Coordinate

Attributes

value : Coded Numeric

A text string representing a date and time specified in the ISO8601 date-time format including timezone designator (TZD): YYYY-MM-DDThh:mm:ssTZD.

1.43 Concept Library::Vendors and Producers

Concepts relating to manufacturers and manufactured things.

1.43.1 Diagram: Vendors and Producers

Figure 137. Vendors and Producers

1.43.2 Class Client <<Role>>

The role of a responsible performer receiving goods or services from a provider.
[FIBO] Client: a party that acquires, or agrees to acquire, ownership (in case of goods), or benefit or usage (in case of services), in exchange for money or other consideration under a contract of sale

1.43.3 Class Individual Product <<Role>>

A specific item or service purchased, sold or offered for sale.
[FIBO] Product: A commercially distributed good that is (1) tangible property, (2) the output or result of a fabrication, manufacturing, or production process, or (3) something that passes through a distribution channel before being consumed or used.

Direct Supertypes

Actual Entity

1.43.4 Class Manufactured Thing <<Role>>

Role of a thing as being made or manufactured.

Direct Supertypes

Individual Product, Item

Attributes

revision : Primitive Value

The revision of a product or good.

1.43.5 Class Producer <<Role>>

Maker of goods or products, usually for sale. Syn: Manufacturer.
[FIBO] Producer: the manufacturer of a product, also called maker.

Direct Supertypes

Supplier

1.43.6 Class Product Kind

A set of similar items or services produced by or delivered by suppliers.

Direct Supertypes

Entity Type

1.43.7 Association Class Product of Supplier <<Relationship>>

A kind of product or service offered by a supplier.

Figure 138. Product Line of Supplier

Direct Supertypes

Impact

Association Ends

has product : Product Kind [0..*]

Product line provided by a manufacturer.

product of : Supplier [1..*]

Manufacturer of a product line

1.43.8 Association Class Production <<Relationship>>

The making of a <produced> thing <produced by> a producer.

Figure 139. Production

Direct Supertypes

Actual Occurrence, Impact

Association Ends

produced by : Producer [1..*]

Entity which manufactured or created an item.

produced : Manufactured Thing [0..*]

A products or good made by a manufacturer.

1.43.9 Association Class Providing <<Relationship>>

The providing of a product or service to a consumer.
[FIBO] supplies

Figure 140. Providing

Direct Supertypes

Actual Occurrence, Impact

Association Ends

provided by : Supplier [0..*]

Supplier of a product or service.
[FIBO] isSuppliedBy

provided to : Client [0..*]

Consumer who receives a product or service.

1.43.10 Class Serial Number <<Value>>

An identifier of an item provided by its producer or supplier.

Direct Supertypes

Managed Item Identifier, Unique Text Identifier

1.43.11 Class Supplier <<Role>>

A person or organization that sells products or services.
[FIBO] a party that supplies goods or services

Direct Supertypes

Social Agent

2 Threat-risk-conceptual-model

The threat risk model

2.1.1 Diagram: Conceptual Threat/Risk Model Diagrams

Figure 141. Conceptual Threat/Risk Model Diagrams

2.2 Threat-risk-conceptual-model::.Summary Diagrams

2.2.1 Diagram: Countermeasures

Figure 142. Countermeasures

2.2.2 Diagram: Entity Kinds

Figure 143. Entity Kinds

2.2.3 Diagram: indicator Fundamentals

Figure 144. indicator Fundamentals

2.2.4 Diagram: MEta Concepts

Figure 145. MEta Concepts

2.2.5 Diagram: Objectives

Figure 146. Objectives

2.2.6 Diagram: Risk Fundamentals

Figure 147. Risk Fundamentals

2.2.7 Diagram: Risk Relationships

Figure 148. Risk Relationships

2.2.8 Diagram: Risk Treatment Fundamentals

Figure 149. Risk Treatment Fundamentals

2.2.9 Diagram: Threat Actor

Figure 150. Threat Actor

2.2.10 Diagram: Vulnerability Essentials

Figure 151. Vulnerability Essentials

2.2.11 Diagram: What is risk?

Figure 152. What is risk?

2.3 Threat-risk-conceptual-model::Threat and Risk Specific Concepts

The risk and threat modules use and specialize more generic concepts to build the risk and threat information sharing and analytics framework.

All risks and threats involve an undesirable situation that is a real or possible *situation* with *consequences* that do harm and impact the *objectives* of *stakeholders*. The same situation may, of course, not be considered a risk or threat to other stakeholders - some may consider such a situation an objective.

This foundational information is then expanded with metrics and interrelationships such that threats and risks can be fully understood and dealt with.

Fundamental risk/threat specific concepts include:

- Danger
- Risk
- Incident
- Indicator
- Vulnerability

Note that these concepts use and build on more generic concepts that are not risk/threat specific such as "person", "organization", and "Intent".

All undesirable situations derived from the general concept of a "situation"; which is fundamental to this specification.

2.3.1 Diagram: Threat and Risk Specific Concepts

Figure 153. Threat and Risk Specific Concepts

2.4 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Accusations

2.4.1 Diagram: Accusations

Figure 154. Accusations

2.4.2 Association Class Accusation <<Relationship>>

An assertion that a perpetrator perpetrated a disruptive action.

Direct Supertypes

Assertion

Association Ends

makes accusaton : Perpetrate [0..*] *Redefines*: negated within: Context

An assertion by a social agent (person or organization) that a threat actor perpetrated some disruptive action.

accusation made by : Social Agent [0..*] *Redefines*: negated within: Context

Social agent asserting that a threat actor perpetrated a disruptive action.

2.5 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Campaigns

Campaigns are ongoing activities in an organized and active way realizing a particular objective of stakeholders.

2.5.1 Diagram: Campaign

Figure 155. Campaign

2.5.2 Diagram: Campaign Objectives

Figure 156. Campaign Objectives

2.5.3 Class Campaign

Campaigns are ongoing work in an organized and active way toward a particular goal, typically a political, military, or social one. A campaign will typically have parts that are the specific activities of the campaign.

A Military campaign is a series of military operations intended to achieve a particular objective, confined to a particular area, or involving a specified type of fighting.

A campaign is also an objective for the activities supporting the campaign.

Direct Supertypes

Objective, Program

2.6 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Danger

Concepts relative to threats. A threat is a situation that increases the likelihood of one or more related incidents.

2.6.1 Diagram: Danger

Figure 157. Danger

2.6.2 Class Attack <<Role>>

A dangerous activity that makes use of and derives benefit from a vulnerability and damages resources.

Direct Supertypes

Capability, Disruptive Action, Threat

2.6.3 Class Indirect Threat <<Role>>

A threat that does directly cause harm but may lead to other situations that will ultimately cause harm.

Direct Supertypes

Threat

2.6.4 Class Natural Threat <<Role>>

A threat from natural means.

Direct Supertypes

Threat

2.6.5 Association Class Target of Attack <<Relationship>>

Target of attack relates an attack to the resources attacked, the attack targets.

Figure 158. Target of Attack

Direct Supertypes

Impact

Association Ends

attack target : Resource [1..*] *Redefines*: negated within: Context

A resource intended to be harmed by an attack.

attacked by : Attack [0..*] *Redefines*: negated within: Context

Attack on a resource.

2.6.6 Class Threat <<Role>>

A threat is role of a situation that may lead to one or more related incidents or failures.

The threat consists of the existence of zero or more threat actors together with a set of one or more vulnerabilities. Thus, the threat of theft may result in an actual theft (attack), and threats correspond to attacks that are typically classified by attacker motivation (e.g., theft) as opposed to technique (e.g., spoofing). In some books and articles, the different but highly related terms “attack” and “threat” are sometimes confounded by being used as synonyms [Firesmith 03, Tulloch 03].

Figure 159. Threat Detail

Direct Supertypes

Dangerous Event

2.6.7 Class Unintentional Threat <<Role>>

A threat that is natural or not intended to cause harm.

Direct Supertypes

Threat

2.7 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Danger Sources

A package categorizing source of dangers - natural, systematic, or intentional

2.7.1 Diagram: Danger Sources

Figure 160. Danger Sources

2.7.2 Class Accident <<Role>>

An unfortunate incident that happens unexpectedly and unintentionally, typically resulting in damage or injury. e.g. a child falling into an open well in a playground.

Direct Supertypes

Capability, Dangerous Event

2.7.3 Association Class Contribution to Danger <<Relationship>>

A relationship defining the undesirable situations a danger source contributes to.

Figure 161. Contribution to Danger

Direct Supertypes

Impact

Association Ends

contributor : Danger Source [0..*] *Redefines*: negated within: Context

A danger source that can contribute to the possibility of an undesirable situation occurring.

contributes to : Undesirable Situation [1..*] *Redefines*: negated within: Context

Undesirable situation that is enabled by a danger source. e.g., an open well contributes to the danger of a child falling in that well.

Attributes

likelihood : Probability Metric

2.7.4 Class Danger Source <<Role>>

The source of any danger - natural, systematic, or intentional, where a danger is a situation (including events) that may lead to an incident that causes harm.

Direct Supertypes

Entity

2.7.5 Class Dangerous Condition <<Role>>

A condition (not an event) that may directly or indirectly lead to harm. e.g. an open well in a playground.

Direct Supertypes

Danger Source, Undesirable Condition

2.7.6 Class Dangerous Event <<Role>>

An event that is the source of danger - natural, systematic, or intentional, where a danger is a that may lead to an incident that causes harm.

Direct Supertypes

Danger Source, Undesirable Event

2.7.7 Association Class Exploit of Vulnerability <<Relationship>>

Vulnerabilities that are leveraged by a danger source to cause or potentially cause harm.

Figure 162. Exploits Vulnerability

Direct Supertypes

Impact

Association Ends

exploits : Vulnerability [1..*] *Redefines*: negated within: Context

Vulnerability used by a danger source (intentionally or unintentionally) to directly or indirectly cause harm.

exploited by : Danger Source [0..*] *Redefines*: negated within: Context

Danger source that can or did exploit a vulnerability such that it leads to an undesirable situation.

2.7.8 Class Objective to Disrupt

Something that efforts or actions are intended to attain such that it damages another in some way or obtains resources not intended for a threat actor.

Direct Supertypes

Objective

2.7.9 Class Unwitting Participant <<Role>>

An actor facilitating an activity or process without their prior knowledge or consent.

Direct Supertypes

Actor, Danger Source

2.8 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Danger Sources::Source of Danger Categories

2.8.1 Diagram: Source of Danger Categories

Figure 163. Source of Danger Categories

2.8.2 Class Biological Danger <<Category>>

Any danger from a biological source.

Direct Supertypes

CBRN Danger

2.8.3 Class CBRN Danger <<Category>>

A Chemical, biological, radiological or nuclear danger.

Direct Supertypes

Source of Danger Category

2.8.4 Class Chemical Danger <<Category>>

A danger from a chemical.

Direct Supertypes

CBRN Danger

2.8.5 Class Civil Unrest Danger <<Category>>

Danger resulting from the actions of a group of people causing disruption of the normal course of society.

Direct Supertypes

Security Danger

2.8.6 Class Criminal Danger <<Category>>

Danger from a criminal activity.

Direct Supertypes

Security Danger

2.8.7 Class Cyber Danger <<Category>>

Danger related to computer systems and networks.

Direct Supertypes

Security Danger

2.8.8 Class Fire Danger <<Category>>

Danger from fire

Direct Supertypes

Source of Danger Category

2.8.9 Class Geophysical Danger <<Category>>

A danger resulting from the geography or movement of the earth.

Direct Supertypes

Source of Danger Category

2.8.10 Class Loss of Control Danger <<Category>>

Categorization of a danger resulting from the control of a system gained by parties not intended to have that control.

Direct Supertypes

Source of Danger Category

2.8.11 Class Meteorological Danger <<Category>>

Category of meteorological impact (e.g., flood).

Direct Supertypes

Geophysical Danger

2.8.12 Class Nuclear Danger <<Category>>

Category of danger from a nuclear blast.

Direct Supertypes

CBRN Danger

2.8.13 Class Radiological Danger <<Category>>

A categorization of danger from radiation.

Direct Supertypes

CBRN Danger

2.8.14 Class Safety Danger <<Category>>

General emergency and public safety danger category. [CAP]

Direct Supertypes

Source of Danger Category

2.8.15 Class Security Danger <<Category>>

[CAP] Danger category of "Security" - Law enforcement, military, homeland and local/private security.

Direct Supertypes

Source of Danger Category

2.8.16 Class Source of Danger Category <<Category>>

A categorization of the source of dangers.

Direct Supertypes

Danger Source

2.8.17 Class Terrorism Danger <<Category>>

Category of danger from terrorism.

Direct Supertypes

Security Danger

2.8.18 Class War Danger <<Category>>

A Category of danger from acts of war.

Direct Supertypes

Security Danger

2.9 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents

Concepts relating to incidents - undesired events that actually happen and impact victims.

2.9.1 Diagram: Incident

Figure 164. Incident

2.9.2 Association Class Danger Leads to Incident <<Relationship>>

The relationship between an incident and its causes, dangerous events.

Figure 165. Cause of Incident

Direct Supertypes

Cause and Effect, Contribution to Danger, Dangerous Event

Association Ends

leads to : Incident [0..*] *Redefines*: negated within: Context

Incident that is the result of a threat.

caused by : Dangerous Event [0..*] *Redefines*: negated within: Context

Cause of an incident.

2.9.3 Class Incident <<Role>>

An incident is a dangerous situation that is happening or has happened directly causing harm (detriment) to victims. Kinds of incidents include attacks, disasters, and accidents. Incidents are actualized risks.
[NIEM] IncidentType

Direct Supertypes

Actual Occurrence, Dangerous Event

2.9.4 Class Witness <<Role>>

Role of a person who observes an event, typically a crime or accident, take place.

Direct Supertypes

Observer

2.9.5 Association Class Witnessing <<Relationship>>

Witnessing is the observation of an incident by a witness

Figure 166. Witnessing

Direct Supertypes

Observation

Association Ends

witnesses : Incident [1..*] *Redefines*: negated within: Context

Incident observed by a witness.

witnessed by : Witness [0..*] *Redefines*: negated within: Context

Witnesses of an incident

2.10 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Failure

2.10.1 Diagram: Failure

Figure 167. Failure

2.10.2 Class Failure

Failure is an incident which causes a resource to not fulfill its intended function.

Direct Supertypes

Incident

2.10.3 Association Class Failure of Resource <<Relationship>>

Relationship between a failure event end the resources that have failed.

Figure 168. Failure Of Resource

Direct Supertypes

Impact

Association Ends

has failure : Failure [0..*] *Redefines*: negated within: Context

Realized or potential failures of a resource.

failure of : Resource [1..*] *Redefines*: negated within: Context

Resource that fails in a failure event. The resource is no longer able to fulfill its function as intended.

2.11 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Failure Categories

2.11.1 Diagram: Failure Categories

Figure 169. Failure Categories

2.11.2 Class Access Control Failure <<Category>>

Failure of permission controls to prevent unintended access.

Direct Supertypes

Control Failure

2.11.3 Class Control Failure <<Category>>

Failure of a policy or control process.

Direct Supertypes

Failure Category

2.11.4 Class Cyber System Failure <<Category>>

Failure of any cyber system - hardware, software, or network to operate as intended.

Direct Supertypes

System Failure

2.11.5 Class Failure Category <<Category>>

A category of failure of a resource.

Direct Supertypes

Failure

2.11.6 Class Industrial Control Failure <<Category>>

A category of danger to the automated control of industrial systems.

Direct Supertypes

Cyber System Failure, Physical System Failure

2.11.7 Class Physical System Failure <<Category>>

A category of danger of failure of any physical system.

Direct Supertypes

System Failure

2.11.8 Class Process Failure <<Category>>

Categorization of a failure of a process to fulfill its objectives.

Direct Supertypes

Failure Category

2.11.9 Class System Failure <<Category>>

Category of failure of a system - physical, financial, cyber, etc. such that the system is no longer available to serve its objectives.

Direct Supertypes

Failure Category

2.12 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Impact Categories

2.12.1 Diagram: Impact Categories

Figure 170. Impact Categories

2.12.2 Class Communications Impact <<Category>>

Category of impact to the ability to communicate.

Direct Supertypes

Information Impact

2.12.3 Class Compliance Impact <<Category>>

Impact on the ability to comply with policies.

Direct Supertypes

Non-Technical Impact

2.12.4 Class Control of Information Impact <<Category>>

Information becomes available to unauthorized actors (loss of confidentiality).

Direct Supertypes

Information Impact

2.12.5 Class Decision-making Impact <<Category>>

Impact on the ability to make informed decisions.

Direct Supertypes

Non-Technical Impact

2.12.6 Class Disinformation Impact <<Category>>

Impact resulting from incorrect information.

Direct Supertypes

Information Impact

2.12.7 Class Electromagnetic Spectrum Impact <<Category>>

Impact based on disruption in spectrum.

Direct Supertypes

Impact Category

2.12.8 Class Environmental Impact <<Category>>

A danger to the environment.

Direct Supertypes

Impact Category

2.12.9 Class Financial Impact <<Category>>

Impact resulting in loss of capital or the ability to obtain capital.

Direct Supertypes

Impact Category

2.12.10 Class Health Impact <<Category>>

A category of danger to people's health.

Direct Supertypes

Impact Category

2.12.11 Class Identity Spoof Impact <<Category>>

A category of danger where an individuals' identity is assumed by another. .

Direct Supertypes

Disinformation Impact

2.12.12 Class Image Impact <<Category>>

Impact to how an entity is viewed by others.

Direct Supertypes

Non-Technical Impact

2.12.13 Class Impact Category <<Category>>

Categorization of the impact of dangers. Danger categories may be combined.

Direct Supertypes

Consequence

2.12.14 Class Information Availability Impact <<Category>>

A category of danger impact where information becomes unavailable to the information owner.
Availability: Ensuring timely and reliable access to and use of information.
[NIST800]

Direct Supertypes

Information Impact

2.12.15 Class Information Impact <<Category>>

A category of danger impact related to information - its unauthorized use or modification.

Direct Supertypes

Impact Category

2.12.16 Class Infrastructure Impact <<Category>>

Classification of impact to infrastructure such that it is not longer available to fulfill objectives.

Direct Supertypes

Impact Category

2.12.17 Class Intellectual Property Impact <<Category>>

Category indicating a loss or compromise of intellectual property.

Direct Supertypes

Non-Technical Impact

2.12.18 Class Legal Impact <<Category>>

A categorization of impact to legal status or legal measures.

Direct Supertypes

Non-Technical Impact

2.12.19 Class Mission Impact <<Category>>

Category of the impact on the ability to achieve a mission purpose.

Direct Supertypes

Impact Category

2.12.20 Class Non-Technical Impact <<Category>>

Category representing the impact to something other than the ability to operate.

Direct Supertypes

Impact Category

2.12.21 Class Safety Impact <<Category>>

Category of impact to the safety of a resource.

Direct Supertypes

Non-Technical Impact

2.12.22 Class Transport Impact <<Category>>

[CAP] A category of impact to public or private transportation.

Direct Supertypes

Impact Category

2.13 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Incidents::Incident Response

2.13.1 Diagram: Incident Response

Figure 171. Incident Response

Incident recovery is an incident response that restores the functionality of impacted resources and systems, including the removal of temporary containment measures.

2.13.2 Class Incident Containment <<Role>>

Containment is an incident response that has two major goals: stopping the spread of the dangers causing the incident and preventing further harm. Containment includes the eradication of latent vulnerabilities of and infections to systems, such as the removal of malware.

Direct Supertypes

Incident Response

2.13.3 Class Incident Recovery <<Role>>

Direct Supertypes

Incident Response

2.13.4 Class Incident Response <<Role>>

Incident response is an activity to address and manage the scope and impact of an incident.

Direct Supertypes

Response

2.14 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Indicators

Indicators are patterns of *situations* or lists of entities to watch out for that **indicate** a **situation** that may happen. An indicator may be scoped by an entity/situation, which contextualizes when it applies. When a situation matching an indicator is *observed* there is a *sighting* of that indicator which is then *evidence* for an *actual situation*, such as an *incident*. e.g., watch for these terrorists in airports.
A *sighting* is an *observation* that matches the pattern of an indicator. e.g., the terrorist Killer-Joe was seen at BWI on 12/11/2014 by a police officer Sam Shoe.
Indicators are not certain and may have a likelihood attached to the potential situations for which they are evidence. Once a sighting flags an indicator, a *course of action* rule may be fired based on the indicated situation.
Specializations of indicator include indicator patterns (an arbitrary pattern of anything) and watch lists. Indicators may also be grouped.

2.14.1 Diagram: Indicator

Figure 172. Indicator

2.14.2 Diagram: Observability

Figure 173. Observability

2.14.3 Diagram: Sighting

Figure 174. Sighting

2.14.4 Class Blacklist Indicator

A list of watched entities that are assumed to pose a threat.

Direct Supertypes

Indicator Watchlist

2.14.5 Class Indicator

An **indicator is a predicate such that** matching sightings are evidence for a possible situation. Sightings matching an indicator suggests further study or action based on a **course of action rule**.
Subtypes of indicator define what is watched for: a pattern, a watch list or some combination of other indicators. Indicators may be restricted to (relevant only within) a specific scope.

Direct Supertypes

Resource, Situation Type

2.14.6 Association Class Indicator Indicates Situation <<Relationship>>

Relationship between an indicator that classifies some kind of situation and the kind of potential situation that the indicator suggests. The indicator is evidence for the indicated situation.

Figure 175. Indicator Indicates Situation

Direct Supertypes

Temporally Related

Association Ends

indicates situation : Potential Situation [1..*] *Redefines*: negated within: Context

The situation pattern that may be inferred by a sighting matching the subject indicator.

pattern indicated by : Indicator [*] *Redefines*: negated within: Context

Indicator that may be used as evidence that a potential situation has been realized by an actual situation.

Attributes

likelihood : Probability Metric

The probability that the indication relationship actually indicates the situation. Note that likelihood may be negative, indicating that the indicator suggests the absence of the indicated pattern .

negation : Boolean

[STIX] The negate field specifies the absence of the indicated pattern, inverting the sign of "likelihood".

2.14.7 Class Indicator Pattern

An indicator defined by a pattern. When the pattern is matched, the indicator "fires", indicating instances of the indicated potential situations.

Direct Supertypes

Indicator, Pattern

2.14.8 Class Indicator Watchlist

An indicator defined by a set of entities (which can be individuals' or situations) that should be watched for in a particular context. The members of the watch list are represented by "watches". When a watched individual is sighted, the indicator "fires", indicating instances of the indicated potential situations.

Direct Supertypes

Indicator

2.14.9 Association Class Observability <<Relationship>>

Observability is a relationship representing the capability of an <can be utilized by> actor to observe an <has ability to utilize> indicator.

Direct Supertypes

Ability

Association Ends

: Indicator [*] *Redefines*: negated within: Context

: Observer [*] *Redefines*: negated within: Context

2.14.10 Association Class Scope of Indicator <<Relationship>>

Relationship defining the scope of an indicators validity, the indicator is only valid within this context.

Figure 176. Scope of Indicator

Direct Supertypes

Extent of Context

Association Ends

has indicator : Indicator [*] *Redefines*: negated within: Context

Indicators that influence the evaluation of the state of an entity.

has scope : Entity [1..*] *Redefines*: negated within: Context

An entity that defines a context for where an indicator is valid.

2.14.11 Association Class Sighting <<Relationship>>

A sighting is a kind of observation that matches one or more indicators and that by matching indicators it increases the likelihood of the situation it <indicates>.

Sightings of indicators are not absolute. Both the matching of the sighting to the indicator and the indicated situation are qualified with a likelihood.

Figure 177. Sighting Detail

Direct Supertypes

Observation

Association Ends

: Entity [*] *Redefines*: negated within: Context

: Observer [0..*] *Redefines*: negated within: Context

2.14.12 Association Class Sighting Indicates Situation <<Relationship>>

Sightings provide evidence that the <indicates> situation may be occurring. This is based on the probability metric of the indicator the sighting is an instance of. Sighting Indicates Situation is derived from Indicator Indicates Situation where each potential situation indicated becomes a possible actual situation as evidenced by the sighting.

Similar to [SACM] AssertedEvidence: The AssertedEvidence association class records the declaration that one or more artefacts of Evidence (cited by ArtefactElementCitations) provide information that helps establish the truth of a Claim. {AssertedEvidence references the record (artefacts) where as sightings are the actual event which may have record}

Figure 178. Sighting Indicates Situation

Direct Supertypes

Temporally Related

Association Ends

indicates : Situation [*] *Redefines*: negated within: Context

Situation that is indicated by a sighting, the sighting is providing evidence for the situation.

indicated by : Sighting [*] *Redefines*: negated within: Context

Sightings that have been observed with respect to an actual situation, providing evidence of the situation.

Attributes

likelihood : Probability Metric

Probability that the specific sighting indicates the specific, actual situation referenced. In many cases this will be the same as the likelihood of the indicator's "Indicator Indicates Situation" likelihood, but specific conditions may increase or decrease that likelihood.

2.14.13 Association Class Sighting Matches Indicator <<Relationship>>

An assertion that a observation matches a particular indicator, thus classifying the observation as a sighting and providing evidence for situations the indicator, indicates.

Figure 179. Sighting Matches Indicator

Direct Supertypes

Extent of Type

Association Ends

matches indicator : Indicator [1..*] *Redefines*: negated within: Context

Indicator the subject sighting matches.The subject sighting will then provide evidence for an actual situation that is an instance of the potential situation of the indicator.

has sighting : Sighting [*] *Redefines*: negated within: Context

Sightings that match the subject indicator and may therefore imply the indicated situation.

Attributes

confidence : Confidence Metric

Confidence that the observed sighting does actually match the pattern. e.g. The witness is 50% certain that the man in the airport had a gun.

2.14.14 Association Class Watch <<Relationship>>

Relationship defining that <watches> is on the <watched by> watch list.

Figure 180. Watch

Direct Supertypes

Temporally Related

Association Ends

watches : Entity [1..*] *Redefines*: negated within: Context

The entities (individuals', systems, situations, etc.) watched by a watch list.

watched by : Indicator Watchlist [*] *Redefines*: negated within: Context

Watch list the subject entity is listed in.

2.14.15 Class Whitelist Indicator

A list of watched entities that are assumed not to pose a threat.

Direct Supertypes

Indicator Watchlist

2.15 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Risk

Concepts relative to risk and risk analytics where risk is an assessment of the potential harm to resources important to a risk owner. At a high level risk is computed as the sum of the likelihood * impact for all resources valued by a risk owner.

2.15.1 Diagram: Risk

Figure 181. Risk

2.15.2 Diagram: Risk Metrics

Figure 182. Risk Metrics

2.15.3 Diagram: Risk Objectives

Figure 183. Risk Objectives

2.15.4 Association Class Impose Strategy <<Relationship>>

A risk owner imposes a risk treatment strategy as a policy with the intent that the strategy reduces their risk to an acceptable level. The policy will be imposed in the entity that the policy <constrains>.

Figure 184. Impose Strategy

Direct Supertypes

Assertion of Policy

Association Ends

imposes : Risk Treatment Strategy [*] *Redefines*: negated within: Context

Risk treatment strategy imposed by a risk owner to mediate the owners risk.

imposed by : Risk Owner [*] *Redefines*: negated within: Context

Authority that imposes a risk strategy such that it is intended to reduce their risk.

2.15.5 Class Objective forSafety and Security

Classification of an objective related to the safety and security of a stakeholder. The protection from harm.

Direct Supertypes

Risk Reduction Objective

2.15.6 Class Objective to Protect Assets

Objective of a stakeholder to protect assets. To reduce the potential harm to valued assets.

Direct Supertypes

Risk Reduction Objective

2.15.7 Class Risk

[CNSSI 4009] Risk is a measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of Event.

[Note: Information system-related security risks are those risks that arise from the loss of confidentiality, integrity, or availability of information or information systems and reflect the potential adverse impacts to organizational operations (including mission, functions, image, or reputation), organizational assets, individuals', other organizations, and the Nation.]

Direct Supertypes

Desirability Assessment

2.15.8 Class Risk Mitigation Strategy

A plan which minimizes or eliminates the possibility or impact of a danger or risk.

Direct Supertypes

Risk Treatment Strategy

2.15.9 Class Risk Owner <<Role>>

A stakeholder with an objective to manage risk.
[ISO 73:2009] person or entity with the accountability and authority to manage a risk.

Figure 185. Risk Owner

Direct Supertypes

Authority, Social Agent, Stakeholder

2.15.10 Class Risk Reduction Objective

An objective of a risk owner to reduce risk.

Direct Supertypes

Objective

2.15.11 Association Class Risk To Resource <<Relationship>>

The Risk to Resource relationship identifies the resources for which risk will be measured for the risk owner.

Figure 186. Risk To Resource

Direct Supertypes

Entity Assessment

Association Ends

measures risk to : Valued Asset [1..*] *Redefines*: negated within: Context

Resources (aka assets) at risk.

has risk of harm : Risk [0..*] *Redefines*: negated within: Context

Potential risk to the subject resource.

2.15.12 Association Class Risk Topic <<Relationship>>

One or more undesirable situations that are assessed together in terms of their risk - the impact and likelihood that those situations will cause harm to resources valued by a risk owner.

Figure 187. Risk Topic

Direct Supertypes

Entity Assessment

Association Ends

measures risk of : Undesirable Situation [1..*] *Redefines*: negated within: Context

Undesirable situations measured in terms of their risk of harming resources valued by a risk owner.

may entail risk : Risk [0..*] *Redefines*: negated within: Context

Risk resulting from a situation happening where the situation may cause harm to resources valued by a risk owner.

2.15.13 Association Class Stakeholder Risk <<Relationship>>

A relationship representing the impact of risk owned by a risk owner. The impact may be ranked as part of risk assessment.

Figure 188. Stakeholder Risk

Direct Supertypes

Entity Assessment, Impact

Association Ends

risk for : Risk Owner [1..*] *Redefines*: negated within: Context

Owner of a risk. The risk owner has objectives to minimize the subject risk.

has risk to objectives : Risk [1..*] *Redefines*: negated within: Context

Risk owned by a risk owner such that the risk may impact the risk owners objectives.

Attributes

rank : Integer

An ordering of how important a risk is relative to all the risks of a risk stakeholder. How the rank is computed is usually determined by net risk but is not specified in this specification.

2.15.14 Class Threat Likelihood <<Quantity Kind>>

A metric representing the likelihood of a threat occurring.

Direct Supertypes

Probability Metric

2.15.15 Association Class Valuation of Asset <<Relationship>>

A relationship representing the set of valued assets for a stakeholder's objectives.

Association Ends

values : Valued Asset [*] *Redefines*: negated within: Context

An asset for which there is an objective to create, sustain, or protect the asset.

supported by : Objective to Protect Assets [1..*] *Redefines*: negated within: Context

An objective that supports the creation, sustainment, or safety of a valued asset.

Attributes

sensitivity threshhold : Metric [0..1]

A metric representing the threshold over which damage to an asset will be of concern to a risk stakeholder.

2.15.16 Class Valued Asset <<Role>>

A system, organization, thing, process or person that is the direct concern of a stakeholder.
[BMM] Asset: something of value owned by the enterprise

Direct Supertypes

Resource

Attributes

replacement cost : Currency [0..1]

Cost to replace the capability offered by the subject valued asset. This may or may not be the cost to replace the asset with an identical one.

2.15.161 Enumeration Likelihood Categories

A high-level scale of likelihood.

Direct Known Superclasses

Scale

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Risk
public enum Likelihood Categories
{Frequent, Probable, Occasional, Remote, Improbable}

Literals

Frequent

Likely to occur often in the life of an item, with a probability of Event greater than 10:1 in that life.

- Probable

Will occur several times in the life of an item, with a probability of Event less than 10:1 but greater than 10:2 in that life.

- Occasional

Likely to occur sometime in the life of an item, with a probability of Event less than 10-2 but greater than 10:3 in that life.

- Remote

Unlikely but possible to occur in the life of an item, with a probability of Event less than 10:3 but greater than 10:6 in that life.

- Improbable

So unlikely, it can be assumed Event may not be experienced, with a probability of Event less than 10:6 in that life.

2.15.162 Enumeration Severity Categories

A high-level scale of severity.

Direct Known Superclasses

Scale

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Risk
public enum Severity Categories
{Catastrophic, Critical, Marginal, Negligable}

Literals

- Catastrophic

Could result in death, permanent total disability, loss exceeding $1M, or irreversible severe environmental damage that violates law or regulation.

- Critical

Could result in permanent partial disability, injuries, or occupational illness that may result in hospitalization of at least three personnel, loss exceeding $200K but less than $1M, or reversible environmental damage causing a violation of law or regulation.

- Marginal

Could result in injury or occupational illness resulting in one or more lost work day(s), loss exceeding $10K but less than $200K, or mitigatible environmental damage without violation of law or regulation where restoration activities can be accomplished.

- Negligable

Could result in injury or illness not resulting in a lost work day, loss exceeding $2K but less than $10K, or minimal environmental damage not violating law or regulation.

Known other enumerations

Enumeration Likelihood Categories, Enumeration Severity Categories

Known other enumerations

Enumeration Likelihood Categories, Enumeration Severity Categories

2.16 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Risk Treatments

Concepts relative to risk treatment. Risk treatments lessen the likelihood or impact of undesirable situations.

2.16.1 Diagram: Risk Treatment

Figure 189. Risk Treatment

2.16.2 Diagram: Safeguard Monitoring

Figure 190. Safeguard Monitoring

2.16.3 Class Accept Risk

A strategy to live with the consequences of a risk and not mitigate it.

Direct Supertypes

Risk Treatment Strategy

Attributes

risk level accepted : Metric

A metric representing the level of acceptable risk for the accept risk strategy.

2.16.4 Association Class Assume Risk <<Relationship>>

A relationship defining the stakeholder assuming a risk for another as part of a risk transfer risk strategy.

Figure 191. Assume Risk

Direct Supertypes

Impact

Association Ends

transfer risk to : Risk Agent [1..*] *Redefines*: negated within: Context

Stakeholder that assumes a risk as the result of a transfer risk action.

assumes risk from : Transfer Risk [1..*] *Redefines*: negated within: Context

The stakeholder that assumes a risk for another, such as an insurance company.

2.16.5 Class Avoid Danger

A likelihood reduction strategy whereby a stakeholder decides not to engage in a risky activity.

Direct Supertypes

Risk Treatment Strategy

2.16.6 Class Countermeasure <<Role>>

Countermeasure – actions, devices, procedures, techniques, or other measures that are designed to lower the likelihood of a threat scenario succeeding in its objectives, reduce the likely consequences of a threat scenario, or minimizing vulnerabilities that a threat scenario can exploit.
Derived from [ISO 28001:2015], [ISO/IEC 2382:2015]

Related concept: Security controls are safeguards or countermeasures to avoid, detect, counteract, or minimize security risks to physical property, information, computer systems, or other assets.

Direct Supertypes

Resource, Situation

2.16.7 Association Class Countermeasure for Strategy <<Relationship>>

Countermeasure for strategy defines a specific <leverages countermeasure> countermeasure that helps to mitigate risk as part of a <countermeasure for> risk mitigation strategy.

Figure 192. Countermeasure for Strategy

Direct Supertypes

Usage

Association Ends

leverages countermeasure : Countermeasure [1..*] *Redefines*: negated within: Context

Countermeasure which serves a risk mitigation strategy.

countermeasure for : Risk Mitigation Strategy [*] *Redefines*: negated within: Context

Mitigation strategy supported by a countermeasure.

2.16.8 Association Class Countermeasure Mitigates <<Relationship>>

Undesirable situations mitigated by a countermeasure.

Figure 193. Countermeasure Mitigation

Direct Supertypes

Impact

Association Ends

reduce harm via : Countermeasure [*] *Redefines*: negated within: Context

An actual or potential response to a danger to minimize the impact of the subject undesirable situation.

mitigates : Undesirable Situation [1..*] *Redefines*: negated within: Context

Undesirable situation for which mitigation reduces the likelihood or impact.

2.16.9 Class Defensive Step

A situation realizing a performance goal that is an objective to protect some valued asset.
Derived from [Schweitzer 2013] Defense node

Direct Supertypes

Scenario Step

2.16.10 Class Management Control

Derived from [NIST.IR.7298]: The security controls (i.e., safeguards or countermeasures) that focus on the management of risk and the management of information system security.
SOURCE: SP 800-37; SP 800-53; SP 800-53A; FIPS 200

Direct Supertypes

Countermeasure

2.16.11 Class Mitigation Actor <<Role>>

Actor that performs a mitigation.

Direct Supertypes

Actor, Resource

2.16.12 Association Class Monitor <<Relationship>>

Monitor relates the <monitored by> safeguard with a <watch based on> indicator such that the monitoring of the indicator becomes a safeguard that <protects> resources.

Figure 194. Monitor

Direct Supertypes

Usage

Association Ends

watch based on : Indicator [0..*] *Redefines*: negated within: Context

Indicators watched by the subject safeguard.

monitored by : Monitoring Control [0..*] *Redefines*: negated within: Context

Activities monitoring an indicator using a monitoring safeguard.

2.16.13 Class Monitoring Control

The action or process of observing something or someone based on well-defined indicators so as to mitigate risks.

Direct Supertypes

Operational Control

2.16.14 Class Operational Control

[NIST.IR.7298] The security controls (i.e., safeguards or countermeasures) for an information system that primarily are implemented and executed by people (as opposed to systems).
SOURCE: SP 800-53; SP 800-37; FIPS 200

Direct Supertypes

Activity, Countermeasure

2.16.15 Association Class Protection <<Relationship>>

The protection of a resource by a countermeasure.

Figure 195. Protection

Direct Supertypes

Resource Dependency

Association Ends

protects : Resource [1..*] *Redefines*: negated within: Context

Resource protected by a countermeasure such that the risk to the resource is reduced. Note that resources include actors, people, organizations and things.

protected by : Countermeasure [*] *Redefines*: negated within: Context

Countermeasure that protects the subject resource.

2.16.16 Class Risk Agent <<Role>>

An entity that assumes risk on behalf of another. e.g. an insurance company.

Direct Supertypes

Risk Owner

2.16.17 Association Class Risk Treatment <<Relationship>>

Risk treatment connects a <modified by> risk treatment strategy with the <modifies risk> Risk that it is intended to reduce.

Figure 196. Risk Treatment

Direct Supertypes

Impact

Association Ends

modifies risk : Risk [1..*] *Redefines*: negated within: Context

Risk that a risk treatment strategy reduces for the risk owner.

modified by : Risk Treatment Strategy [0..*] *Redefines*: negated within: Context

A strategy to reduce the subject risk for the risk owner.

2.16.18 Class Risk Treatment Strategy

A plan, method or process for dealing with risk by reducing the likelihood or impact.

Direct Supertypes

Policy

Attributes

degree of mitigation : Metric

A metric for how much a mitigation reduces the likelihood or impact of an undesirable situation.

2.16.19 Association Class Safeguarding <<Relationship>>

The safeguarding relationship relates a <mitigated by> mitigation actor with a <performs safeguard> safeguard activity such that it <protects> resources.

Figure 197. Safeguarding

Direct Supertypes

Performance

Association Ends

performs safeguard : Operational Control [*] *Redefines*: negated within: Context

An activity a mitigation actor performs to protect resources identified by <protects>.

mitigated by : Mitigation Actor [*] *Redefines*: negated within: Context

The actor(s) that perform an activity to safeguard resources.

2.16.20 Class Technical Control

Derived from [NIST.IR.7298]: The security controls (i.e., safeguards or countermeasures) that are primarily implemented and executed by the information system through mechanisms contained in the hardware, software, or firmware components of the system.
SOURCE: SP 800-53; SP 800-53A; SP 800-37; FIPS 200

Direct Supertypes

Countermeasure

2.16.21 Class Transfer Risk

A strategy to cause another to assume the impact of a risk. e.g., insurance.

Direct Supertypes

Risk Treatment Strategy

2.17 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Threat Actors

2.17.1 Diagram: Threat Actors

Figure 198. Threat Actors

2.17.2 Class Disrupt Stakeholder's Objective

An action intended to harm the objectives of a stakeholder.

Direct Supertypes

Capability, Disruptive Action

2.17.3 Class Disruptive Action <<Role>>

An intentional activity that serves the objectives of a threat actor to cause harm or violate law. Some disruptive actions are threats.

Figure 199. Disruptive Actions

Direct Supertypes

Activity, Dangerous Event

2.17.4 Class Disruptive Step

A situation realizing a performance goal that is an objective to disrupt some valued asset.
Derived from [Schweitzer 2013] Attack node

Direct Supertypes

Scenario Step

2.17.5 Association Class Perpetrate <<Relationship>>

An actor involved in perpetrating an attack or other disruptive action.

Figure 200. Perpetrate

Direct Supertypes

Contribution to Danger, Performance

Association Ends

perpetrates : Disruptive Action [*] *Redefines*: negated within: Context

The activity performed by a threat actor to cause or contribute to an attack.

perpetrator : Threat Actor [1..*] *Redefines*: negated within: Context

The threat actor performing an activity to cause or contribute to an attack.

2.17.6 Class Threat Actor <<Role>>

Role of an actor; all or partially responsible for some undesired situation - threat, risk, or attack. Threat actors have intent to do harm.

Direct Supertypes

Actor, Danger Source, Stakeholder

2.18 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Undesirable Situations

Undesirable situations are a fundamentally concept that unifies the threat-risk framework. Undesirable situations classify a situation as one that causes harm to stakeholders (directly or indirectly). Undesirable situations are further specialized across three dimensions:

- Events Vs. Conditions - Events "happen" whereas conditions are a steady state for some period.
- Actual Vs. Potential.
- Intentional Vs. Natural or Systematic. Intentional dangers involve a "threat actor" whereas unintentional only involve weaknesses in resources.

The above are used to define more specific risk & threat concepts, such as:

- Incidents which are actual dangerous situations.
- Disasters and Accidents which are unintentional actual situation (no threat actor).
- Attacks which are actual situations perpetrated by a threat actor.
- Risks which are potential dangerous situations, thus having some level of uncertainty.
- Threats which are intentional risks from a threat actor.
- Hazards which are natural or systematic risks.

Resources also play an important role in the risk/threat framework in that resources are harmed by dangers but risks are also important for attackers to exploit vulnerabilities and for defenders to realize mitigations.

2.18.1 Diagram: Undesirable Situations

Figure 201. Undesirable Situations

2.18.2 Class Harm

Harm is a consequence of a situation that negatively impacts the objectives of stakeholders and therefore has negative desirability for those stakeholders.

[Firesmith 2003] Harm is a negative impact associated with an asset. Harm is due to an accident when dealing with safety requirements, is due to an attack when dealing with security requirements, and may be due to both accidents and attacks when dealing with survivability requirements.

[NIEM] Injury (More specific concept - Person specific).

Figure 202. Harm

Direct Supertypes

Consequence, Undesirable Situation

2.18.3 Association Class Harms Resource <<Relationship>>

Resources harmed or potentially harmed by an undesirable situation.

Figure 203. Harms Resource

Direct Supertypes

Harm, Impact

Association Ends

harms : Resource [*] *Redefines*: negated within: Context

harmed by : Undesirable Situation [*] *Redefines*: negated within: Context

Attributes

likelihood : Probability Metric

2.18.4 Association Class Harms Victim <<Relationship>>

People or organizations (social agents) harmed or potentially harmed by an undesirable situation.

Figure 204. Harms Victim

Direct Supertypes

Harms Resource, Involvement

Association Ends

harms victim : Victim [*] *Redefines*: negated within: Context

Victim harmed by a situation.

victim of : Undesirable Situation [1..*] *Redefines*: negated within: Context

Situations for which the subject is a victim.

2.18.5 Association Class Source of Harm <<Relationship>>

Relationship describing the harm produced as a result of a situation.

Figure 205. Harm Source

Direct Supertypes

Consequence of Situation

Association Ends

causes harm : Harm [*] *Redefines*: negated within: Context

The harm to a resource caused by a undesirable situation.

harm from : Undesirable Situation [1..*] *Redefines*: negated within: Context

Situation which contributes to harm.

2.18.6 Class Undesirable Condition <<Role>>

A role of a situation as an undesirable static situation (a condition, not something happening) that directly or indirectly does or may have detrimental consequences impacting the objectives of stakeholders.

Direct Supertypes

State, Undesirable Situation

2.18.7 Class Undesirable Event <<Role>>

A role of an Event that may directly or indirectly cause harm.

Direct Supertypes

Occurrence, Undesirable Situation

2.18.8 Class Undesirable Situation <<Role>>

An undesirable situation is a role of a situation (condition or event) that has, is, or may cause harm (directly or indirectly). Undesirable situations negatively impact the objectives of stakeholders. An undesirable situation may be classified in the context of the impacted stakeholders - what is undesirable to one stakeholder may be desirable to another.

Direct Supertypes

Situation

Attributes

severity : Metric

A metric for the total harm caused by a undesirable situation.

2.18.9 Class Victim <<Role>>

The role of any actor harmed by an incident.

Direct Supertypes

Social Agent, Valued Asset

2.19 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities

Vulnerabilities and weaknesses represent flaws or inherent qualities of some resource that can be the source of danger.

2.19.1 Diagram: Vulnerability

Figure 206. Vulnerability

2.19.2 Diagram: Vulnerability Identifiers

Figure 207. Vulnerability Identifiers

2.19.3 Class Physical Vulnerability

A category of vulnerability of something to physical danger or attack.

Direct Supertypes

Vulnerability

2.19.4 Association Class Supporting Condition <<Relationship>>

Vulnerabilities required to exist for a threat to transition to an incident.

Figure 208. Supporting Condition

Direct Supertypes

Cause and Effect

Association Ends

condition for : Undesirable Situation [0..*] *Redefines*: negated within: Context

Incident or failure for which a vulnerability is a condition.

has condition : Vulnerability [0..*] *Redefines*: negated within: Context

A vulnerability as a condition for the a undesirable condition to occur.

Attributes

required : Boolean Value

True if the condition is required for the undesirable condition to occur, false if the condition is one of many conditions that may enable the undesirable situation.

2.19.5 Class Vulnerability

A Vulnerability (of an object and a cause of failure, i.e. attack, natural cause, mistake, natural cause, accidental cause, or indirect) is the set of conditions under which an object fails under the particular cause of failure.
This is consistent with NIST 800-30 (based on CNSSII 4009)
Vulnerability is a weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source.

Direct Supertypes

Undesirable Condition

Attributes

previously known : Boolean

At the time of the latest vulnerability characterization, true if the vulnerability had been previously reported, false if newly discovered or "zero day". Time is recorded in the "starts on" property of vulnerability.

publicly known : Boolean

An assertion that the existence of vulnerability is public knowledge, not closely held within an organization.

score : Vulnerability Metric

Score for the severity of the subject vulnerability - may use CVSS or other metrics.

vector : Vulnerability Vector [*]

Factors influencing the subject vulnerabilitie's score.

2.19.6 Class Vulnerability Identifier <<Value>>

An identifier for a vulnerability.

Direct Supertypes

Text Identifier

2.19.7 Class Vulnerability Metric <<Quantity Kind>>

A metric representing the overall impact of a vulnerability.

Direct Supertypes

Harm-Benefit Metric

2.19.8 Association Class Vulnerability of Resource <<Relationship>>

Relationship defining the resources that have a particular vulnerability.

Figure 209. Vulnerability of Resource

Direct Supertypes

Impact

Association Ends

vulnerability of : Resource [1..*] *Redefines*: negated within: Context

Resource the subject vulnerability may impact.

has vulnerability : Vulnerability [0..*] *Redefines*: negated within: Context

Vulnerabilities of a resource, ways it may be compromised.

2.20 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Cyber Vulnerabilities

2.20.1 Diagram: Cyber Vulnerability

Figure 210. Cyber Vulnerability

2.20.2 Class Communications Vulnerability

A potential failure of a communications systems to restrict information flow to unintended parties.

Direct Supertypes

Cyber Vulnerability

2.20.3 Class CVE Identifier <<Value>>

An identifier for Common Vulnerabilities and Exposures [https://cve.mitre.org/].

Direct Supertypes

Vulnerability Identifier

2.20.4 Class Cyber Vulnerability

A vulnerability of any cyber related resource.

Direct Supertypes

Vulnerability

2.20.5 Class Information System Vulnerability

Category of vulnerability of a computer system and/or its network, software and execution of processes.

Direct Supertypes

Cyber Vulnerability

2.20.6 Class Information Vulnerability

A category of vulnerability of information loss, misuse or corruption.

Direct Supertypes

Cyber Vulnerability

2.20.7 Class OSVDB Identifier <<Value>>

OSVDB is an independent and open sourced web-based vulnerability database created for the security community.[http://osvdb.org/]. This package defines identifiers for referencing OSVDB.

Direct Supertypes

Vulnerability Identifier

Attributes

value : Integer [1]

Index into the OSVDB database.

2.20.8 Class Software Vulnerability

A vulnerability of software such that the software such that the softwares operation is comprised.

Direct Supertypes

Cyber Vulnerability

2.21 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors

[cvss] IT management must identify and assess vulnerabilities across many disparate hardware and software platforms. They need to prioritize these vulnerabilities and remediate those that pose the greatest risk. But when there are so many to fix, with each being scored using different scales, how can IT managers convert this mountain of vulnerability data into actionable information? The Common Vulnerability Scoring System (CVSS) is an open framework that addresses this issue. It offers the following benefits:

- Standardized Vulnerability Scores: When an organization normalizes vulnerability scores across all of its software and hardware platforms, it can leverage a single vulnerability management policy. This policy may be similar to a service level agreement (SLA) that states how quickly a particular vulnerability must be validated and remediated.
- Open Framework: Users can be confused when a vulnerability is assigned an arbitrary score. “Which properties gave it that score? How does it differ from the one released yesterday?” With CVSS, anyone can see the individual characteristics used to derive a score.
- Prioritized Risk: When the environmental score is computed, the vulnerability now becomes contextual. That is, vulnerability scores are now representative of the actual risk to an organization. Users know how important a given vulnerability is in relation to other vulnerabilities.

2.21.1 Diagram: Vulnerability Vectors

Figure 211. Vulnerability Vectors

2.21.2 Class CVSS Score <<Quantity Kind>>

Common Vulnerability Scoring System. A number in the range of 0..10 representing a CVSS metric.
[CVSS]

Direct Supertypes

Vulnerability Metric

2.21.21 Enumeration Access Complexity

[CVSS] This metric measures the complexity of the attack required to exploit the vulnerability once an attacker has gained access to the target system. For example, consider a buffer overflow in an Internet service:
once the target system is located, the attacker can launch an exploit at will.
Other vulnerabilities, however, may require additional steps in order to be exploited. For example, a vulnerability in an email client is only exploited after the user downloads and opens a tainted attachment.
The lower the required complexity, the higher the vulnerability score.

Direct Known Superclasses

Vulnerability Base Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Access Complexity
{High, Medium, Low}

Literals

High

Specialized access conditions exist. For example:

- In most configurations, the attacking party must already have elevated privileges or spoof additional systems in addition to the attacking system (e.g., DNS hijacking).
- The attack depends on social engineering methods that would be easily detected by
- knowledgeable people. For example, the victim must perform several suspicious or
- atypical actions.
- The vulnerable configuration is seen very rarely in practice.
- If a race condition exists, the window is very narrow.

Medium

The access conditions are somewhat specialized; the following are examples:

- The attacking party is limited to a group of systems or users at some level of authorization, possibly untrusted.
- Some information must be gathered before a successful attack can be launched.
- The affected configuration is non-default, and is not commonly configured (e.g., a vulnerability present when a server performs user account authentication via a specific scheme, but not present for another authentication scheme).
- The attack requires a small amount of social engineering that might occasionally fool cautious users (e.g., phishing attacks that modify a web browser's status bar to show a false link, having to be on someone's "buddy" list before sending an IM exploit).

Low

Specialized access conditions or extenuating circumstances do not exist. The following are examples:

- The affected product typically requires access to a wide range of systems and users, possibly anonymous and untrusted (e.g., Internet-facing web or mail server).
- The affected configuration is default or ubiquitous.
- The attack can be performed manually and requires little skill or additional information gathering.

- The “race condition” is a lazy one (i.e., it is technically a race but easily winnable).

2.21.22 Enumeration Access Vector

[CVSS]This metric reflects how the vulnerability is exploited. The more remote an attacker can be to attack a host, the greater the vulnerability score.

Direct Known Superclasses

Vulnerability Base Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Access Vector
{Local, Adjacent, Remote}

Literals

Local

Local access to a vulnerable resource.
[cvss]A vulnerability exploitable with only local access requires the attacker to have either physical access to the vulnerable system or a local (shell) account. Examples of locally exploitable vulnerabilities are peripheral attacks such as Firewire/USB DMA attacks, and local privilege escalations (e.g., sudo).

Adjacent

A resource vulnerable to an attacker with have access adjacent to the vulnerable resource.
[cvss] Adjacent Network. A vulnerability exploitable with adjacent network access requires the attacker to have access to either the broadcast or collision domain of the vulnerable software. Examples of local networks include local IP subnet, Bluetooth, IEEE 802.11, and local Ethernet segment.

Remote

A vulnerability that does not require physical or virtual proximity.
[cvss] Network: A vulnerability exploitable with network access means the vulnerable software is bound to the network stack and the attacker does not require local network access or local access.
Such a vulnerability is often termed “remotely exploitable”. An example of a network attack is an RPC buffer overflow.

2.21.23 Enumeration Authentication

[CVSS] This metric measures the number of times an attacker must authenticate to a target in order to exploit a vulnerability. This metric does not gauge the strength or complexity of the authentication process, only that an attacker is required to provide credentials before an exploit may occur. The fewer authentication instances that are required, the higher the vulnerability score.
It is important to note that the Authentication metric is different from Access Vector. Here, authentication requirements are considered once the system has already been accessed. Specifically, for locally exploitable vulnerabilities, this metric should only be set to “single” or “multiple” if authentication is needed beyond what is required to log into the system. An example of a locally exploitable vulnerability that requires authentication is one affecting a database engine listening on a Unix domain socket (or some other non-network interface). If the user must authenticate as a valid database user in order to exploit the vulnerability, then this metric should be set to “single.”

Direct Known Superclasses

Vulnerability Base Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Authentication
{Multiple, Single, None}

Literals

- Multiple

Exploiting the vulnerability requires that the attacker authenticate two or more times, even if the same credentials are used each time. An example is an attacker authenticating to an operating system in addition to providing credentials to access an application hosted on that system.

- Single

One instance of authentication is required to access and exploit the vulnerability.

- None

Authentication is not required to access and exploit the vulnerability.

2.21.24 Enumeration Availability Impact

[CVSS] Impact affecting the availability of a resource for its intended use.

Direct Known Superclasses

Vulnerability Base Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Availability Impact
{None, Partial, Complete}

Literals

- None

[cvss] There is no impact to the availability of the system.

- Partial

[cvss] There is reduced performance or interruptions in resource availability. An example is a network-based flood attack that permits a limited number of successful connections to an Internet service.

- Complete

[cvss] There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.

2.21.25 Enumeration Collateral Damage Potential

[CVSS] This metric measures the potential for loss of life or physical assets through damage or theft of property or equipment. The metric may also measure economic loss of productivity or revenue. Naturally, the greater the damage potential, the higher the vulnerability score.

Direct Known Superclasses

Vulnerability Environmental Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Collateral Damage Potential
{None, Low, Low-Medium, Medium High, High}

Literals

- None

[cvss] There is no potential for loss of life, physical assets, productivity, or revenue.

- Low

[cvss] A successful exploit of this vulnerability may result in slight physical or property damage. Or, there may be a slight loss of revenue or productivity to the organization.

- Low-Medium

[cvss] A successful exploit of this vulnerability may result in moderate physical or property damage. Or, there may be a moderate loss of revenue or productivity to the organization.

- Medium High

[cvss] A successful exploit of this vulnerability may result in significant physical or property damage or loss. Or, there may be a significant loss of revenue or productivity.

- High

[cvss] A successful exploit of this vulnerability may result in catastrophic physical or property damage and loss. Or, there may be a catastrophic loss of revenue or productivity.

2.21.26 Enumeration Confidentiality Impact

[CVSS] This metric measures the impact on confidentiality of a successfully exploited vulnerability.
Confidentiality refers to limiting information access and disclosure to only authorized users, as well as preventing access by, or disclosure to, unauthorized ones. Increased confidentiality impact increases the vulnerability score.

Direct Known Superclasses

Vulnerability Base Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Confidentiality Impact
{None, Partial, Complete}

Literals

- None

[cvss] There is no impact to the confidentiality of the system.

- Partial

[cvss] There is considerable informational disclosure. Access to some system files is possible, but the attacker does not have control over what is obtained, or the scope of the loss is constrained. An example is a vulnerability that divulges only certain tables in a database.

- Complete

[cvss] There is total information disclosure, resulting in all system files being revealed. The attacker is able to read all of the system's data (memory, files, etc.)

2.21.27 Enumeration Exploitability

This metric measures the current state of exploit techniques or code availability. Public availability of easy-to-use exploit code increases the number of potential attackers by including those who are unskilled, thereby increasing the severity of the vulnerability.
Initially, real-world exploitation may only be theoretical. Publication of proof of concept code, functional exploit code, or sufficient technical details necessary to exploit the vulnerability may follow. Furthermore, the exploit code available may progress from a proof-of-concept demonstration to exploit code that is successful in exploiting the vulnerability

consistently. In severe cases, it may be delivered as the payload of a network-based worm or virus. The more easily a vulnerability can be exploited, the higher the vulnerability score.

Direct Known Superclasses

Vulnerability Temporal Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Exploitability
{Unproven, Proof of concept, Functional, High}

Literals

- Unproven

[cvss] No exploit [code] is available, or an exploit is entirely theoretical.

- Proof of concept

[cvss] Proof-of-concept exploit [code] or an attack demonstration that is not practical for most systems is available. The code or technique is not functional in all situations and may require substantial modification by a skilled attacker.

- Functional

[cvss] Functional exploit [code] is available. The [code/exploit] works in most situations where the vulnerability exists.

- High

[cvss] Either the vulnerability is exploitable by functional mobile autonomous code, or no exploit is required (manual trigger) and details are widely available. The code works in every situation, or is actively being delivered via a mobile autonomous agent (such as a worm or virus).

2.21.28 Enumeration Integrity Impact

[CVSS] This metric measures the impact to integrity of a successfully exploited vulnerability. Integrity refers to the trustworthiness and guaranteed veracity of information. Increased integrity impact increases the vulnerability score.

Direct Known Superclasses

Vulnerability Base Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Integrity Impact
{None, Partial, Complete}

Literals

- None

[cvss] There is no impact to the integrity of the system.

- Partial

Control over the system is partially comprised.
[cvss] Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited. For example, system or application files may be overwritten or modified, but either the attacker has no control over which files are affected or the attacker can modify files within only a limited context or scope.

- Complete

[CVSS] There is a total compromise of system integrity. There is a complete loss of system protection, resulting in the entire system being compromised.

2.21.29 Enumeration Remediation Level

[CVSS] A way to express the degree of remediation that can be provided.

Direct Known Superclasses

Vulnerability Temporal Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Remediation Level
{Offical Fix, Temporary Fix, Workaround, Unavailable}

Literals

- Offical Fix

[cvss] A complete vendor solution is available. Either the vendor has issued an official patch, or an upgrade is available.

- Temporary Fix

[cvss] There is an official but temporary fix available. This includes instances where the vendor issues a temporary hotfix, tool, or workaround.

- Workaround

[cvss] There is an unofficial, non-vendor solution available. In some cases, users of the affected technology will create a patch of their own or provide steps to work around or otherwise mitigate the vulnerability.

- Unavailable

[cvss] There is either no solution available or it is impossible to apply

2.21.210 Enumeration Report Confidence

[CVSS] Confidence in a report.

Direct Known Superclasses

Vulnerability Temporal Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Report Confidence
{Unconfirmed, Uncorroborated, Confirmed}

Literals

- Unconfirmed

[cvss] There is a single unconfirmed source or possibly multiple conflicting reports. There is little confidence in the validity of the reports. An example is a rumor that surfaces from the hacker underground.

- Uncorroborated

[cvss] There are multiple non-official sources, possibly including independent security companies or research organizations. At this point there may be conflicting technical details or some other lingering ambiguity.

- Confirmed

[cvss] The vulnerability has been acknowledged by the vendor or author of the affected technology. The vulnerability may also be “Confirmed” when its existence is confirmed from an external event such as publication of functional or proof-of-concept exploit code or widespread exploitation.

2.21.211 Enumeration Security Requirements

[CVSS] These metrics enable the analyst to customize the CVSS score depending on the importance of the affected IT asset to a user's organization, measured in terms of confidentiality, integrity, and availability. That is, if an IT asset supports a business function for which availability is most important, the analyst can assign a greater value to availability, relative to confidentiality and integrity. Each security requirement has three possible values: "low," "medium," or "high." The full effect on the environmental score is determined by the corresponding base impact metrics. That is, these metrics modify the environmental score by reweighting the (base) confidentiality, integrity, and availability impact metrics. For example, the confidentiality impact (C) metric has increased weight if the confidentiality requirement (CR) is "high." Likewise, the confidentiality impact metric has decreased weight if the confidentiality requirement is "low." The confidentiality impact metric weighting is neutral if the confidentiality requirement is "medium." This same logic is applied to the integrity and availability requirements.
Note that the confidentiality requirement will not affect the environmental score if the (base) confidentiality impact is set to "none." Also, increasing the confidentiality requirement from "medium" to "high" will not change the environmental score when the (base) impact metrics are set to "complete."
This is because the impact sub score (part of the base score that calculates impact) is already at a maximum value of 10. The greater the security requirement, the higher the score (remember that "medium" is considered the default).

Direct Known Superclasses

Vulnerability Environmental Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Security Requirements
{Low, Medium, High}

Literals

- Low

[cvss] Loss of [confidentiality | integrity | availability] is likely to have only a limited adverse effect on the organization or individuals' associated with the organization (e.g., employees, customers).

- Medium

[cvss] Loss of [confidentiality | integrity | availability] is likely to have a serious adverse effect on the organization or individuals' associated with the organization (e.g., employees, customers).

- High

[cvss] Loss of [confidentiality | integrity | availability] is likely to have a catastrophic adverse effect on the organization or individuals' associated with the organization (e.g., employees, customers)

2.21.212 Enumeration Target Distribution

[CVSS] This metric measures the proportion of vulnerable systems. It is meant as an environment-specific indicator in order to approximate the percentage of systems that could be affected by the vulnerability.
The greater the proportion of vulnerable systems, the higher the score.

Direct Known Superclasses

Vulnerability Environmental Vector

package Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Vulnerabilities::Vulnerability Vectors
public enum Target Distribution
{None, Low, Medum, High}

Literals

- None

[cvss] No target systems exist, or targets are so highly specialized that they only exist in a laboratory setting. Effectively 0% of the environment is at risk.

- ○ Low

[cvss] Targets exist inside the environment, but on a small scale. Between 1% - 25% of the total environment is at risk.

- ○ Medum

[cvss] Targets exist inside the environment, but on a medium scale. Between 26% - 75% of the total environment is at risk

- ○ High

[cvss] Targets exist inside the environment on a considerable scale. Between 76% - 100% of the total environment is considered at risk.

2.22 Threat-risk-conceptual-model::Threat and Risk Specific Concepts::Weapons

2.22.1 Diagram: Weapons

Figure 212. Weapons

2.22.2 Class Cyber Weapon <<Role>>

A software weapon able to exploit the vulnerabilities of a cyber system.

Direct Supertypes

Cyber Resource, Weapon

2.22.3 Class Physical Weapon <<Role>>

A physical item intended to exploit a vulnerability and cause harm to some class of entities.

Direct Supertypes

Physical Tool, Weapon

2.22.4 Class Weapon <<Role>>

Role of something used by an actor to cause harm by exploiting vulnerabilities.

Direct Supertypes

Tool

2.22.5 Association Class Weapon Leverages Vulnerability <<Relationship>>

Relationship defining the weapons that can leverage (exploit) a vulnerability.

Figure 213. Weapon Leverages Vulnerability

Direct Supertypes

Impact

Association Ends

leverages : Vulnerability [0..*] *Redefines*: negated within: Context

Vulnerability that a weapon helps an actor take advantage of.

leveraged by : Weapon [0..*] *Redefines*: negated within: Context

Weapon that can leverage a vulnerability to help an actor take advantage of that vulnerability.

3 SMIF Conceptual Model

3.1 Diagram: SMIF Packages

Figure 214. SMIF Packages

3.2 SMIF Conceptual Model::Associations

An association asserts a formal condition involving related things, the association ends. An association may be asserted within a context as true or false within that context. Each association has a number of bindings of which are immutable for that association.

Associations are differentiated from relationships in that associations are fully dependent on the things they relate. These are known as "formal", "thin", "internal" or "intrinsic" relations in much of the literature.

3.2.1 Diagram: Associations

Figure 215. Associations

3.2.2 Class Association

An association makes a logical statement involving related things, the association ends. An association may be asserted within a context as true or false within that context. Each association type has a number of bindings of which are immutable for that association.
An association may be true or false within its context and is atomic in its truth value.
Associations are differentiated from relationships in that associations are not situations - they are not temporal and do not change over time. Associations may be a consequence of relationships or other situations or may be derived from qualities of associated ends.
Associations can "own" owned property bindings as their "ends".
See also: Relationship
[Guizzardi] Intrinsic Relation
[UML] Link
[DOLCE] Formal Relation
[ORM] Fact

Direct Supertypes

Identifiable Property Owner, Proposition

3.2.3 Class Association Type

A type of Association (See Association for details) which defines a set of "Association Property Types" which are the types of association property bindings. Associations are not situations - the are not temporal things. This does not prevent subtypes of associations from being situations.
[Guizzardi] Intrinsic Relation Type
[UML] Association
[OWL] For binary associations, may be considered a pair of properties that are Inverse Object Properties.
[ORM] Fact Type

Direct Supertypes

Property Owner Type

3.2.4 Class Binary Association

A binary association is an association with exactly 2 property bindings corresponding to exactly 2 owned property types.

Direct Supertypes

Association

Associations

<<Restriction>> : Binary Association Type [1..*] *Subsets*: has type:Type

3.2.5 Class Binary Association Type

A binary association type is an association type having exactly two owned property types (association ends).

Direct Supertypes

Association Type

Associations

<<Restriction>> : Binary Association [0..*] *Redefines*: categorizes:Thing

: Owned Property Type [2] *Redefines*: ownes property type:Owned Property Type

3.2.6 Class Restriction Type

A restriction type restricts another type, frequently an association, by holding only constraints. All properties owned by a restriction type must be property restrictions.

Direct Supertypes

Property Owner Type

3.3 SMIF Conceptual Model::Constraints

Rules define constraints or behaviors that are asserted in specified context.

3.3.1 Diagram: General Rules

Figure 216. General Rules

3.3.2 Diagram: Property Constraints

Figure 217. Property Constraints

This diagram focuses on rules about properties.

3.3.3 Diagram: Rules in Context

Figure 218. Rules in Context

This diagram shows how rules are propositions that may be asserted within any context to apply to any other context, thus realizing the "open world assumption".

3.3.4 Diagram: Rules Summary

Figure 219. Rules Summary

This diagram shown a summary of the primary rules.

3.3.5 Diagram: Type Constraints

Figure 220. Type Constraints

This diagram focuses on rules about types (note that property types are also types).

3.3.6 Class Conditional

Anything with a condition defined by an expression.

Attributes

<<Characteristic>>condition : Expression Node [0..1]
Condition that must be TRUE for an element to be asserted. All values other than "TRUE" are FALSE.

3.3.7 Class Conditional Rule <<Rule>>

A rule with a general expression as a condition that applies to what the rule <constrains>. Where asserted, the condition must be true.
[UML] Constraint where "context" corresponds with <holds within> and "constrainedElement" corresponds with "constrains". "specification" corresponds with "condition".

Direct Supertypes

Conditional, Constraint

3.3.8 Class Constraint <<Rule>>

A constraint is a rule enumerating a specific set of constrained entities as identified by <constrains>.

Direct Supertypes

Identifiable Property Owner, Proposition

3.3.9 Association Constraint Covers

Relationship defining the types covered by a covering constraint.

Association Ends

is covered by : Type [*] *Redefines*: ownes property type: Owned Property Type

A type covered by a covering constraint.

The <constrains> type must be a direct supertype of all <is covered by> types.

has covering : Covering Constraint [*] *Redefines*: ownes property type: Owned Property Type

Covering constraints of a type.

3.3.10 Class Covering Constraint <<Rule>>

A constraint that the extent (<categorizes> things) of the <constrains> type is equivalent to the union of the extents of the <is covered by> types.
[UML] GeneralizationSet with isCovering=TRUE. "constrains" corresponds with the common "general" of each Generalization". "is covered by" corresponds with each "special" of each generalization.

Direct Supertypes

Type Constraint

3.3.11 Class Disjoint <<Rule>>

Disjoint is a rule that the things denoted by what the rule <constrains> do not and may not denote any of the same set of things.
When applied to a context (including types) all elements contextualized are included in the set of disjoint individuals.

[FIBO] Mutually Exclusive sets

[IDEAS] PartitionOfSetOfDisjointIndividuals: A FusionOfSetOfIndividuals whose fusioned Type is a SetOfDisjointIndividuals.

[UML] [UML] GeneralizationSet with isDisjoint=TRUE. "constrains" corresponds with "is covered by" of each "special" of each generalization. Note the SMIF does not require that disjoint elements have a common supertype, one may be inferred for UML mapping.

[OWL: Union(DisjointClasses, DisjointObjectProperties, DisjointDataProperties, DifferentIndividuals)

Direct Supertypes

Constraint

3.3.12 Class Enumerated <<Rule>>

The contextualized elements of the <constrains> context is a closed (enumerated) set, it can not be extended. A.K.A. "Closed World Assumption". Elements may not be asserted by any context other than the one specified in <holds within>.

[FIBO] Selections of Things
[FUML] Wen constraining a type, corresponds with [FUML] "Enumeration". SMIF enumerations are not limited to literals. The "ownedLiteral" corresponds with all elements owned by <holds within>.
[ISO11404] Enumerated: enumerated is a family of datatypes, each of which comprises a finite number of distinguished values having an intrinsic order.
[OWL] ObjectUnitionOf(DataOneOf, ObjectOneOf)

Direct Supertypes

Constraint

3.3.13 Class Equivalent <<Rule>>

Equivalent is a rule that the things the rule <constraints> denote the same set of things. When applied to a context (including types) each thing the context contextualizes is included in the set of equivalent things.

Related to*: [ISO 1087] synonymy: relation between or among terms (3.4.3) in a given language representing the same concept (3.2.1)

Related to*: [ISO 1087] equivalence: relation between designations (3.4.1) in different languages representing the same concept (3.2.1)

* SMIF relates concepts, not terms. synonymy may also be represented by multiple terms for the same concept.

[OWL] Union(SameIndividual, EquivalentClasses, EquivalentObjectProperties, EquivalentDataProperties)

Direct Supertypes

Constraint

3.3.14 Class Multiplicity Constraint <<Rule>>

A Multiplicity constraint constrains the number of bindings <multiplicity of> types (including property types) may have in a particular instance of the constrained type.

For a property type, The number of instances bound to a property for the set of instances bound to <with respect to> shall

be limited by the minimum and maximum number of the multiplicity.

For non-property types, the multiplicity shall apply to the extent of the type as described by <classifies>.

[IDEAS] superSubType

[FUML] MultiplicityElement: Note: Multiplicity Constraint constraining a type has semantics included in to UML MultiplicityElement.

[OWL] Union(ObjectMaxCardinality, ObjectMinCardinality, ObjectExactCardinality, DataMaxCardinality, DataMinCardinality, DataExactCardinality)

Direct Supertypes

Type Constraint

Attributes

minimum number : Integer Value [0..1]
Minimum number in a set as constrained by a multiplicity.
[FUML] MultiplicityElement.lowerValue
[OWL] MinCardinality

maximum number : Integer Value [0..1]
Maximum number in a set as constrained by a multiplicity.
[FUML] MultiplicityElement.upperValue
[OWL] maxCardinality

at once : Boolean Value = true
When at once is true, the constraint applies for each snapshot in time but not across snapshots (e.g. a car can have at most one driver at a time). When at once is false the constraint applies across all time (e.g. a person has exactly one birth mother across all time).

3.3.15 Association Multiplicity Target

Relationship defining the type a multiplicity rule applies to. Note that properties are types and may also have multiplicity constraints.

Direct Supertypes

Rule Constrains

Association Ends

multiplicity of : Type [1] *Redefines*: ownes property type: Owned Property Type

The type or property that is the subject of a multiplicity constraint.

has multiplicity : Multiplicity Constraint [*] *Redefines*: ownes property type: Owned Property Type

Multiplicity constraint of a type or property.

3.3.16 Class Property Constraint <<Rule>>

Abstract supertype for constraints that constrain properties types.

Direct Supertypes

Constraint

3.3.17 Class Property Transitivity Constraint <<Rule>>

A transitive property defined by <constrains> interlinks two individuals A and C whenever it interlinks A with B and B with C for some individual B.
For example "larger than" is transitive in that if Joe is larger than Sue and Sue is Larger then Sam, then Joe is larger than Sam.
[OWL] TransitionObjectProperty

Direct Supertypes

Property Constraint

3.3.18 Association Proposition Subsumption

Relationship defining subsumption of a proposition used as a rule. When a proposition subsumes another the subsumed proposition will not apply (fire) if the <subsumed by> rules applies (fires).

Association Ends

subsumes : Proposition [*] *Redefines*: ownes property type: Owned Property Type

When a rule subsumes another the subsumed rule will not apply (fire) if the <subsumed by> rules applies (fires). Where rules are also patterns, a rule may specialize another which will subsume the specialized rule as well as include the generalized rule parts as parts of the specialized rule.

subsumed by : Proposition [*] *Redefines*: ownes property type: Owned Property Type

When rule is <subsumed by> another the subsumed rule will not apply (fire) if the <subsumed by> rules applies (fires).

3.3.19 Association Rule Constrains

Relationship defining the entity constrained by a rule. Where no constrained entity is specified, all entities are constrained with the scope of <holds within> are constrained.

Association Ends

constrains : Entity [1..*] *Redefines*: ownes property type: Owned Property Type

The entity or entities constrained by a rule.
Where a rule constrains a context, all things contextualized by the context shall be subject to the rule.
Where there are no <constrains> for a rule, the rule applies globally - to the universal context.

constrained by : Constraint [0..*] *Redefines*: ownes property type: Owned Property Type

Rules applying to an entity.

3.3.20 Class Type Constraint <<Rule>>

A constraint of a type, including Relationships types.

Direct Supertypes

Constraint

3.3.21 Association Unique Set

Relationship defining the set of properties that uniquely identify an instance of the constrained type.

Association Ends

has unique : Property Type [1..*] *Redefines*: ownes property type: Owned Property Type

The set of involved properties within a type that uniquely identify an individual.

has uniqueness constraint : Uniqueness Constraint [*] *Redefines*: ownes property type: Owned Property Type

Uniqueness constraints for a property.

3.3.22 Class Uniqueness Constraint <<Rule>>

A constraint that, within the <constrains> type the rule applies to, the set of instances bound to the set of types in the "has unique" relation must be unique and serves to define the "identity" of each individual.
Note: Uniqueness may be used to define a "key".
[OWL] HasKey where CE (subject class expression) is <constrains> and <has unique> is Union(ObjectPropertyExpression, DataPropertyExpression)

Direct Supertypes

Type Constraint

Attributes

is primary identity : Boolean Value

A uniqueness constraint that can be interpreted as a "primary key", the identity of an entity.

3.4 SMIF Conceptual Model::Expressions

Expressions define computations across SMIF models.

3.4.1 Diagram: Expressions

Figure 221. Expressions

Expressions define computations

3.4.2 Class And Function

Direct Supertypes

Binary Boolean Function

3.4.3 Class Binary Boolean Function

A function returning true or false and having two arguments

Direct Supertypes

Binary Function, Boolean Function

3.4.4 Class Binary Function

Direct Supertypes

Expression Node

Attributes

- argument 1 : Expression Node
- argument 2 : Expression Node

3.4.5 Class Boolean Function

A function returning true or false

Direct Supertypes

Expression Node

3.4.6 Class Constant Reference

A calculation that returns a thing identified by <has value>.

[FIBO] Constant
[FUML] LiteralSpecification where subtype of literal is determined by the type of <has value>.
-LiteralInteger->type is Integer or a subtype
-LiteralReal-> type is not integer or a subtype
-LiteralBoolean->type is Boolean
-LiteralString->type is Text

Direct Supertypes

Expression Node

3.4.7 Association Constant Value

Relationship defining a link to a constant value within an expression.

Association Ends

has value : Thing [1] *Redefines*: ownes property type: Owned Property Type

A constant value referenced in an expression.

constant of : Constant Reference [*] *Redefines*: ownes property type: Owned Property Type

Referencing constant expression node.

3.4.8 Class Current Context

A proxy for the context in which the expression is evaluated. A.K.A. "This" or "Self".

Direct Supertypes

Expression Node

3.4.9 Class Equality Function

Returns TRUE if all <has equal> things have the same value or represent the same thing or set of things regardless of how they are represented.
Equality will return TRUE or FALSE.
[ISO11404: EqualityIn every value space there is a notion of equality, for which the following rules hold:
– for any two instances (a, b) of values from the value space, either a is equal to b, denoted a = b , or a is not equal to b, denoted a ≠ b ;
– there is no pair of instances (a, b) of values from the value space such that both a = b and a ≠ b ;
– for every value a from the value space, a = a ;
– for any two instances (a, b) of values from the value space, a = b if and only if b = a ;
– for any three instances (a, b, c) of values from the value space, if a = b and b = c , then a = c . On every datatype, the operation Equal is defined in terms of the equality property of the value space, by:
– for any values a, b drawn from the value space, Equal(a,b) is true if a = b , and false otherwise.

Direct Supertypes

Binary Boolean Function

3.4.10 Class Expression Context

An abstract element defining the static or dynamic evaluation context and resulting type of an expression.
An expression context that is referenced by another expression context inherits the referencing context by default.

Direct Supertypes

Entity

3.4.11 Association Expression Evaluation

Relationship defining the expression that will be evaluated by an evaluation.

Association Ends

evaluates : Expression Node [1] *Redefines*: ownes property type: Owned Property Type

The expression node "head" an evaluation evaluates.

evaluated by : Expression Context [*] *Redefines*: ownes property type: Owned Property Type

Evaluations of an expression node.

3.4.12 Class Expression Node

An abstract class representing the computation of a value which is then bound to the context from which it is called. Each expression node has a type of the most general type it can return.

An expression node may reference other elements. Where the other elements are also expression nodes they will be considered part of the referencing expression and evaluated in the context of that expression.

The set of related expression nodes forms a "tree" for evaluation.

All references to an expression node other than as a "Constant Reference" shall return the result of evaluating the expression node.
An evaluation may be used in place of anything that requires the <resulting type> of the evaluation.

[FIBO] Expression
[UML] Expression

Direct Supertypes

Thing

3.4.13 Class Function Call

An element of an expression that performs some operation based on a function type and produces a result. I.e. plus(a,1). Arguments are bound to the function call via bindings.

Direct Supertypes

Expression Node, Identifiable Property Owner

3.4.14 Association Function Called

Relationship defining the function (a type) called by a function call.

Direct Supertypes

Extent of Type

Association Ends

calls : Function Type [1] *Redefines*: ownes property type: Owned Property Type

Function called

is used by : Function Call [*] *Redefines*: ownes property type: Owned Property Type

Function calls using a function declaration.

3.4.15 Class Function Type

A declaration of a function which performs a calculation on arguments (properties) to produce a result (function result). I.e. the definition of plus(a:Number, b:Number).

Functions are intended to be side-effect free and context free (they only depend on their arguments and don't change anything) but assertions to specify that certain functions are pure may be required,
Note: FUNCTION ARGUMENTS ARE PROPERTIES of the function.

[FUML] Operation where ownedParameter corresponds with <has property> and type corresponds with <resulting type>.

Direct Supertypes

Expression Context, Property Owner Type, Property Type

3.4.16 Class Not Function

A function returning the complement of its argument.

Direct Supertypes

Boolean Function

Attributes

argument : Expression Node

3.4.17 Class Object Operation Type

An operation bound to a specific "receiver" in the "Object Oriented" sense.
[FUML] Operation

Direct Supertypes

Function Type

3.4.18 Association OO Target

Relationship defining the "target" type of an object oriented function.

Association Ends

receiver : Property Type [1] *Redefines*: ownes property type: Owned Property Type

The property that is the receiver of an object operation.

[UML] class (of Operation)

received by : Object Operation Type [*] *Redefines*: ownes property type: Owned Property Type

The Object Operation for which a receiver is defined.

3.4.19 Class Or Function

Direct Supertypes

Binary Boolean Function

3.4.20 Class Traversal

Traversal from the current <evaluates in> context to another across a relation or other structure.

A traversal is a structure such that the structure's bindings may hold other properties of a traversal constant as independent variables where <traverses through> is the dependent variable. The traversal shall be considered to have the type of the relation it is traversing. Traversing binary relations does not require any bindings.

[OWL] ObjectPropertyChain

Direct Supertypes

Expression Node, Identifiable Property Owner

Attributes

traverse to relation : Boolean Value = false

Where traverse to relation is false, the traversal will return the bound element(s) of the <traverses through> property from the current context via any intermediate relationships.

Where traverse to relation is true, the traversal shall return the structure/situation/relationship owning the property binding.

By default, traverse to relation is false.

inverse : Boolean Value = false

Indicates that the traversal is defined based on properties that reference the current context. This results in traversing "backwards" across a property to an inverse property or the relation.

argument : Expression Node

3.4.21 Association Traverse Through

Relationship defining the property of the current context which will be traversed.

Association Ends

traverses through : Property Type [1] *Redefines*: ownes property type: Owned Property Type

Property or properties through which a traversal traverses as the dependent variable(s).

/ traversed by : Traversal [*] *Redefines*: ownes property type: Owned Property Type

Traversals through a property.

3.4.22 Class Xor Function

Direct Supertypes

Binary Boolean Function

3.5 SMIF Conceptual Model::Facets

The facet package defines facets, roles and phases. Types that "mix in" to other types in a specific context or timeframe.

3.5.1 Diagram: Facets

Figure 222. Facets

3.5.2 Class Category Type

A category is a classification or division of people, events or things regarded as having particular shared characteristics. Categorization is typically contextual, potentially transient and may or may not be formally defined.
As with all facets, categories are non-rigid. Something classified by a category must also be classified by an entity type.

Direct Supertypes

Facet Type

3.5.3 Association Facet Constraint

A Facet Constraint specifies a possible a "mix in" or "non rigid" classification of an entity beyond any fundamental entity type. The must common kinds of facets are roles and phases. Note that the UML profile also specifies subtypes of

<<Facet Of>> that are syntactic sugar for facet classification.

The type of individual a facet can apply to may be specified using a facet classification. Note that Facets . A facet classification may be contextual, such as within a relation, situation and/or time frame. Instances may have any number of types and classifications may change over time.

A <<Sufficient>> property is typically used to specify the mediating concept the facet is with respect to.

There are various implementation strategies for facets including multiple classification and dependent objects. SMIF does not commit to any particular implementation strategy.

Direct Supertypes

Generalization

Association Ends

/ facet of : Type [0..1] *Redefines*: ownes property type: Owned Property Type

/ faceted by : Facet Type [0..*] *Redefines*: ownes property type: Owned Property Type

3.5.4 Association Facet Of Entity

Facet of Entity is a first-class relationship defining a facet of an entity. As a relationship the application of the facet to the entity may be time and context specific. The entity will have the type of the identified facet. Properties and relationships specific to the entity exhibiting a facet, if required, may utilize a subtype of Entity Of Facet.
For example, a Person may play the "Parent" role for multiple children and there may be properties and relationships relative to this generic parent role. If it is required to specify properties and relationships specific to the person being a parent of a specific child, a subtype of Facet Of Entity is used.

Direct Supertypes

Extent of Type

Association Ends

/ exhibits facet : Facet Type [*] *Redefines*: ownes property type: Owned Property Type

The facet that an entity assumes when it is the facet of an entity.

[FIBO] (for roles) playsRole

/ exhibited by : Entity [*] *Redefines*: ownes property type: Owned Property Type

Type of the entity exhibiting the facet.

3.5.5 Class Facet Type

A facet is a "mix in" type that defines an aspect of something but does not define the identity or "fundamental" (A.K.A. "Rigid") type of that thing, but some potentially transient role, phase or other way to classify it. Something must have at least one type that is not a facet to define that things identity.
Facets do not define independent identity of the referent but technology implementations may create independent objects to represent a facet.
An instance of a facet must also have a type that is not a facet to provide the identity of the instance.
The type(s) a facet may categorize may be constrained by a Facet Generalization Constraint. E.g. Policeman is a role of a person.

[Guarino1994] Non-Substantial sortal
[Guizzard] Non-Rigid Universal: A universal G is non-rigid iff for a w ∈ W There is an x such that x ∈ extw(G), and there is a w' ∈ W such that x ∉extw' (G)

[SOWA1999] Prehension (Relative

Direct Supertypes

Type

3.5.6 Class Phase Type

A phase (or state) is a static characteristic of something that exists for limited time(s). Something takes on or looses a phase as a result of some event. E,g, Teenager, living, closed invoice.
A Phase is a situation in that there is a situation coincident with each phase.

[Guizzardi] (Phased-Sortal): Let PS be a universal and let S be a substance sortal specialized (restricted by) PS. Now, let extw(~PS) = extw(S) \ extw(PS) be the complement of the extension of PS in world w. In this formula, the symbol \ represents the set theoretical operation of set difference. The universal PS is a phased-sortal iff for all worlds w ∈ W, there is a w' ∈ W such that extw(PS) ∩ extw□(~PS) ≠ ∅

Direct Supertypes

Facet Type

3.5.7 Class Role Type

A role is a facet type that defines a specific purpose or behavior of a class of things. E.g. teacher, policeman, or employer.
[FIBO] Role. Note that partyInRole or thingInRole are implied by classification of a thing.

Direct Supertypes

Facet Type

3.6 SMIF Conceptual Model::Identifiers

Terms and identifiers provide for signs for (ways to identify) anything.

3.6.1 Diagram: Identifier Summary

Figure 223. Identifier Summary

3.6.2 Diagram: Identifiers

Figure 224. Identifiers

An identifier that can be represented as text. The text is in the "value" property.

[IDEAS] Sign: An Individual that signifies a Thing.

3.6.3 Association Identification

Relationship defining an identifier for an entity.

[IDEAS] namedBy: A couple that asserts that a Name describes a Thing.

[ISO 1087] Designation

Association Ends

/ identifies : Thing [1] *Redefines*: ownes property type: Owned Property Type

The entity an identifier identifies.

[FIBO] identifies: is the relationship between something and that which provides a unique reference for it

[ISO 1087] designator: representation of a concept (3.2.1) by a sign which denotes it

/ identified by : Identifier [*] *Redefines*: ownes property type: Owned Property Type

An identifier for an <Entity>.

[FIBO] hasDenotation

3.6.4 Class Identifier <<Value>>

An identifier is any value that is used to distinguish an entity from other entities. Note that any identifier may be contextualized by one or more context, including language context. Identifiers are a "sign" for an identity where identity is an abstraction of individuality that is the basis for identifiers.

[IDEAS] Name: A Representation that identifies a Thing.
[FIBO] Identifier
[CL] Term: expression which denotes an individual, consisting of either a name or, recursively, a function term applied to a sequence of arguments, which are themselves terms

Direct Supertypes

Structured Value

3.6.5 Association Identifier in Namespace

Relationship defining the namespace within which a unique identifier is defined and unique.

[ISO 1087] monosemy: relation between designations (3.4.1) and concepts (3.2.1) in a given language in which one designation only relates to one concept

Direct Supertypes

Extent of Context

Association Ends

/ unique within : Namespace [1] *Redefines*: ownes property type: Owned Property Type

The namespace in which an identifier is defined and has a unique value.

[FUML] memberNamespace

/ scopes identifier : Unique Identifier [0..*] *Redefines*: ownes property type: Owned Property Type

An Identifier defined within the scope of a namespace.

[FUML] member

3.6.6 Class IRI Identifier <<Value>>

A IRI/URI Identifier for an entity, as defined in [RFC3987].

[FIBO] anyURI

Direct Supertypes

Technical Identifier

3.6.7 Class Name <<Value>>

A word or set of words by which a person, animal, place, or thing is known, addressed, or referred to. Names are not necessarily unique.

[IDEAS] Name: A Representation that identifies a Thing.

[CL] Discourse Name

Direct Supertypes

Text Identifier

3.6.8 Class Namespace

A namespace is a context that provides a way to make identifiers unique and identify exactly one entity. For example, the Virginia driver's license division provides unique driver's license numbers.

Similar to [IDEAS] UniqueNamingScheme: A NamingScheme where different Names will not contain tokens of the same Representation Type.
Note: SMIF identifiers are not instances of their namespace.

[FIBO] IdentificationScheme: system for allocating identifiers to objects

[ISO 1087] terminology 1: set of designations (3.4.1) belonging to one special language (3.1.3)

[FUML] Namespace

[CL] Vocabulary

Direct Supertypes

Lexical Context

3.6.9 Association Naming

Relationship defining a human meaningfully name for an entity.

Direct Supertypes

Identification

Association Ends

/ names : Thing [1..*] *Redefines*: ownes property type: Owned Property Type

An entity named by a name.

/ has name : Name [*] *Redefines*: ownes property type: Owned Property Type

A human meaningful name for an entity.

[FIBO] hasName: that by which some thing is known; may apply to anything

[OWL] rdfs:label

3.6.10 Association Prefered Identification

Relationship defining the preferred identifier for an entity.

[ISO 1087] preferred term: term (3.4.3) rated according to the scale of the term acceptability rating (3.4.14) as the primary term for a given concept (3.2.1)

Direct Supertypes

Identification

Association Ends

has preferred : Identifier [0..1] *Redefines*: ownes property type: Owned Property Type

Default identifier to use for an entity.

Where multiple identifiers are preferred in differing context any method for selecting the most preferred identifier is implementation specific and not specified by this standard.

[FUML] NamedElement.name: Note: An Identifier that is <preferred for> an entity is equivalent to the name of a named element.

preferred for : Thing [0..1] *Redefines*: ownes property type: Owned Property Type

The entity an identifier is preferred for.

3.6.11 Class Technical Identifier <<Value>>

A technical identifier is defined within a technical system, information structure or system of systems for references and identity within that system or information element. Such identifiers may have no meaning outside of that system.

Typical technical identifiers include inter document "refs", record numbers, etc. The system should be referenced as the namespace.

Direct Supertypes

Unique Text Identifier

3.6.12 Class Term <<Value>><<Intersection>>

A word, phrase or name used by stakeholders to uniquely identify entities.

[ISO 1087] term: verbal designation of a general concept in a specific subject field.

Direct Supertypes

Name, Unique Text Identifier

3.6.13 Class Text Identifier <<Value>>

A code or other simple value that can be represented as text, identifying something that may or may not be unique. Simple identifiers may be codes, names, numbers or compound values.

[NIEM] IdentificationType (IdentificationID=value)

Direct Supertypes

Identifier

Attributes

value : String Value

Text value of an identifier

3.6.14 Class Unique Identifier <<Value>>

A unique identifier is an entity used to uniquely identify something. The identified thing is referenced by what the identifier <identifies>.
Identifiers are defined and <unique within> a lexical scope as its namespace.
Multiple identifiers may use the same word or text value (or other forms of values) in differing <unique within> namespaces such that the same word may have different meanings in different context.
An entity may have any number of identifiers.

Direct Supertypes

Identifier

3.6.15 Class Unique Text Identifier <<Value><<Intersection>>

An <Identifier> that is represented using text. e.g. a "word", "phrase" or "name".

Direct Supertypes

Text Identifier, Unique Identifier

3.7 SMIF Conceptual Model::Lexical Scope

Lexical scope defines the structure of models and the ownership of model elements.

3.7.1 Diagram: Lexical Scope

Figure 225. Lexical Scope

3.7.2 Class Conceptual Package

A model of a real or possible world as conceived by the model authors.

Direct Supertypes

Package

3.7.3 Association Definition

Relationship defining the set of elements defined within a lexical scope.
[OWL] RDF Graph

Association Ends

defines : Thing [*] *Redefines*: ownes property type: Owned Property Type

A model element defined within a lexical scope.
Definition within a scope does not assert everything within a scope but the lexical scope may be independently asserted, thus asserting what it defines.
[FUML] ownedElement, ownedMember

defined in : Lexical Scope [1] *Redefines*: ownes property type: Owned Property Type

Lexical scope defining model elements.
[UML]owner

3.7.4 Class Include

An "Include" is an external scope that is visible and asserted by the owning lexical scope.
[FUML] PackageImport
[CL] Importation: An importation contains a name. The intention is that the name identifies a piece of Common Logic content represented externally to the text, and the importation re-asserts that content in the text.

Direct Supertypes

Lexical Reference

3.7.5 Class Information Repository

An information repository is a resource that can store (directly or indirectly) information content, including models. Typical information repositories are directories and file servers.

Direct Supertypes

Lexical Context

3.7.6 Class Lexical Context

Lexical context represents a lexical scope that is also a context - asserting the contained elements

Direct Supertypes

Lexical Scope, Scoped Context

3.7.7 Class Lexical Reference

A Lexical Reference is an external scope that is visible to but not necessarily asserted by the owning lexical scope.

Direct Supertypes

Context

3.7.8 Class Lexical Scope

Lexical scope represents model content (the lexical structure of the model) that then models an area of concern. A lexical scope may define model elements representing anything.
[CL] Text: A text is a set, list, or bag of phrases. A piece of text shall optionally be identified by a name.
[OWL] Potential scope of a RDF graph defined by <defines>

Direct Supertypes

Thing

3.7.9 Class Logical Package

A model of information about systems independent of technical representation.

Direct Supertypes

Package

3.7.10 Class Model

A root package. A model has no owner and may be directly referenced as an independent information resource. A model is defined in it's self.

Direct Supertypes

Package

3.7.11 Class Package

A model element that provides a definitional scope for other model elements. A package may be represented as a "graph".

[ISO 1087] concept system: system of concepts set of concepts (3.2.1) structured according to the relations among them

[FUML] Package. FUML ownedMember corresponds with SMIF <defines>. FUML "nestedPackage" corresponds with "defines" where the element defined is a package.

[CL] Module: A module consists of a name, an optional set of names called the exclusion set, and a text called the body text.

Direct Supertypes

Lexical Context

3.7.12 Class Physical Package

A physical, technology specific, data schema representing information about a real or possible world.

Direct Supertypes

Package

3.7.13 Class Prefix <<Value>>

A technical abbreviation for a package.

Direct Supertypes

Unique Text Identifier

3.7.14 Association Prefix Text

Relationship defining the prefix for a package.

Direct Supertypes

Identification

Association Ends

has prefix : Prefix [0..1] *Redefines*: ownes property type: Owned Property Type

An abbreviation that can be used to identify a package.

prefix of : Lexical Scope [1] *Redefines*: ownes property type: Owned Property Type

An abbreviation for a lexical context, including packages and models.

3.7.15 Association Repository Conteining Model

Association Ends

Stores Model : Model [0..*] *Redefines*: ownes property type: Owned Property Type

Stored In : Information Repository [1] *Redefines*: ownes property type: Owned Property Type

3.7.16 Association Scope of Reference

Relationship defining internal or external context that are referenced by a lexical scope using a lexical reference.

Association Ends

References scope : Context [1] *Redefines*: ownes property type: Owned Property Type

A referenced context, potentially in another model, that provides visibility to the elements in that context.
[FUML] importedPackage
[OWL] directlyImports (implies "Include")

referenced by : Lexical Reference [*] *Redefines*: ownes property type: Owned Property Type

References to a context.

3.7.17 Association Scope Reference

Relationship defining references for a scope.

Association Ends

references : Lexical Reference [*] *Redefines*: ownes property type: Owned Property Type

A reference providing visibility of a lexical scope to an internal or external context.

extends scope : Lexical Context [1] *Redefines*: ownes property type: Owned Property Type

A lexical scope that is extended by a lexical reference.
[FUML] importingNamespace

3.7.18 Association Statement

Relationship defining the set of elements defined within and asserted by a lexical scope.

Direct Supertypes

Assertion, Definition

Association Ends

states : Proposition [0..*] *Redefines*: ownes property type: Owned Property Type

<states> combines <defines> with <has assertion> to both define and assert an element within a lexical scope. <states> provides a more "structural" organization of concepts that are both defined and asserted in the same structure.

<states> is a convenience for the common case where assertion and lexical containment are combined.

stated by : Lexical Context [0..1] *Redefines*: ownes property type: Owned Property Type

<stated by> is a lexical scope that both defines and asserts a model element.

3.8 SMIF Conceptual Model::Mapping

Mapping rules define how data represents concepts or how different data representations are related.

3.8.1 Diagram: Facades

Figure 226. Facades

3.8.2 Diagram: Mapping Rules

Figure 227. Mapping Rules

3.8.3 Class Computed Facade

A facade that is computed by calling external methods.

Direct Supertypes

Facade

Operations

- public push ()

An operation called to evoke the behavior associated with a new facade element being created or modified. Push asserts the more concrete type based on a reference type.

- public pull ()

An operation called to evoke the behavior associated with a facade representing existing elements. Pull asserts the reference type based on a more concrete type.

3.8.4 Association Concrete Map End

Relationship to the more concrete end of a match rule.

Association Ends

concrete end : Match End [1] *Redefines*: ownes property type: Owned Property Type

One end of a mapping, to be used for more concrete end.

match from : Match Rule [0..1] *Redefines*: ownes property type: Owned Property Type

Mapping rule owning a "concrete" end.

3.8.5 Association Concrete Pattern Body

Relationship between a mapping and a pattern of the more concrete concepts to be mapped.

Association Ends

concrete focus : Pattern Variable [1] *Redefines*: ownes property type: Owned Property Type

The variable or variables that form the basis for the portion of the pattern for the more concrete (physical) model. The concrete portion of the pattern is derived from the transitive closure of all variables reachable from the pattern variable via characteristics, associations or relationships.
When a pattern matching the set of concrete variables is created or altered the mapping "fires" and the reference pattern is asserted.
The qualification of the referenced variable is constrained to be ""select".

concrete mapping : Mapping [1] *Redefines*: ownes property type: Owned Property Type

Mapping for which a more concrete pattern is defined.

3.8.6 Class Facade

An intermediary data type used to hold common mappings. Facades may be computed and/or have mapping rules.

3.8.7 Association Map Rule Type Assertion

Relationship defining more concrete types that shall be asserted for an end of a match rule.

Association Ends

asserted type : Type [*] *Redefines*: ownes property type: Owned Property Type

Type that will be asserted for the end that is more concrete than the defined type of a property or relationship. e.g. a unit type.

asserted by : Match End [*] *Redefines*: ownes property type: Owned Property Type

Map rule and that asserts a type

3.8.8 Association Mapped variable

Relationship defining the property that is the source or target of a mapping

Association Ends

maps variable : Pattern Variable [1] *Redefines*: ownes property type: Owned Property Type

Variable that defines a set of elements to map to the other side of the mapping rule. The set of elements shall be those bound to the property on evaluation of the mapping.

maps to : Match End [*] *Redefines*: ownes property type: Owned Property Type

Map rule end for a property

3.8.9 Class Mapping <<Rule>>

A mapping is a rule based on a pattern that defines how different representations of the same things correspond. There are two "sub patterns", defined by the concrete and reference variables and other variables reachable from them via characteristics, associations and relationships. These sub-patterns are matched (made to correspond) using "Match Rules"
Patterns define a set of related elements to be mapped based on two distinguished variables, the "concrete body" and the "reference body".
Types in a "concrete" body may be defined to be a representation (data about) a concept in a "reference" pattern.
Match rules define how elements in each of the sub-patterns are mapped, bidirectionally.
A mapping utilizing more specific types subsumes maps for more general types.
Note that the roles of "concrete" and "reference" may or may not reflect different levels of abstraction and in some cases the choice may be arbitrary.

Direct Supertypes

Constraint, Pattern

Attributes

- strength : Assertion Strength

Strength defines what will cause a rule to be considered for being asserted (firing).

3.8.10 Class Match End

One end of a mapping from one thing to another that may be qualified with a condition.
The set of elements to be mapped is the union of the sets of all mapped types and mapped variables that conform to the condition.
Match rules are constrained to apply to only conforming types or types that represent the mapped ends (as specified by a representation rule).
Representation rules applied to a supertype apply to a subtype unless a more specific representation rule is specified for the corresponding types.

Direct Supertypes

Computed, Conditional

3.8.11 Class Match Rule <<Rule>>

A rule that the 2 ends represent the same things or information about a thing.
Redundant mappings are ignored and identity is preserved across all mappings.

Direct Supertypes

Constraint, Proposition

Attributes

- coerce : Boolean Value

Where <coerce> has a value of TRUE a map rule will be evaluated even if the <reference end> is not type compatible with the <concrete end> type.
Where <coerce> is FALSE or unstated a map rule will be evaluated only if the <reference end> is type compatible with the <concrete end> type.
Type compatible shall be defined as one of: Being the same type, <concrete end> being a subtype of <reference end> (as defined by a type generalization rule), <concrete end> being a representation of <reference end> (as defined by a

representation rule).
Representation rules applied to a supertype apply to a subtype.

3.8.12 Association Match Rules

Relationship defining the match rules for a mapping.

Direct Supertypes

Statement

Association Ends

has map rule : Match Rule [*] *Redefines*: ownes property type: Owned Property Type

Map rule that is asserted by a mapping.

map rule of : Mapping [1] *Redefines*: ownes property type: Owned Property Type

Mapping containing a map rule.

3.8.13 Association Reference Map End

Relationship to the reference end of a match rule.

Association Ends

reference end : Match End [1] *Redefines*: ownes property type: Owned Property Type

One end of a match rule, to be used for more abstract end.

match to : Match Rule [0..1] *Redefines*: ownes property type: Owned Property Type

Mapping rule owning a reference" end.

3.8.14 Association Reference Pattern Body

Relationship between a mapping and a pattern of the more abstract concepts to be mapped.

Association Ends

reference focus : Pattern Variable [1] *Redefines*: ownes property type: Owned Property Type

The variable or variables that form the basis for the portion of the pattern for the more abstract/reference (conceptual) model. The reference portion of the pattern is derived from the transitive closure of all variables reachable from the pattern variable via characteristics, associations or relationships.
When a pattern matching the set of reference variables is created or altered the mapping "fires" and the concrete pattern is asserted.
The qualification of the referenced variable is constrained to be ""select".

reference mapping : Mapping [1] *Redefines*: ownes property type: Owned Property Type

Mapping for which a more abstract pattern is defined.

3.8.15 Association Representation

More concrete type that represents information about the represented concept of a representation rule.

Association Ends

represented by : Type [1] *Redefines*: ownes property type: Owned Property Type

The representation of a concept in a more specific form

represents rule : Representation Rule *Redefines*: ownes property type: Owned Property Type

Rule defining a representation of a type.

3.8.16 Class Representation Rule <<Rule>>

A representation rule states that the <represented type> has a representation defined by the <represented by> type.
Representation rules are used to filter Map Rules such that only represented concepts may be mapped.
A representation is usually complimented with one or more mapping rules.

Direct Supertypes

Conditional Rule

Attributes

map all : Boolean Value

Specifies a direct mapping between instances of the types in both directions.
<map all> is equivalent to a mapping with a rule mapping properties of each type but is lower precedence than other mappings - if types have a more specific map it will apply first.

3.8.17 Association Represented Concept

More abstract type that is <represented by> a more concrete type of a representation rule.

Association Ends

represented type : Type [1..*] *Redefines*: ownes property type: Owned Property Type

A more general or abstract concept that is being represented.

concept rule : Representation Rule *Redefines*: ownes property type: Owned Property Type

Rule defining a concept that is represented by another, more concrete, concept.

3.8.171 Enumeration Assertion Strength

Rule strength defines what will cause a rule to be considered for being asserted (firing).
package SMIF Conceptual Model::Mapping
public enum Assertion Strength
{Global, Local}

Literals

Global

The rule will be in effect globally.

Local

The rule will only be in effect if required to fulfill another rule.

Known other enumerations

Enumeration Assertion Strength

3.9 SMIF Conceptual Model::Metadata

Metadata defines data about model elements (their source, definition or trust), which can be differentiated from model elements about the subject domain.

3.9.1 Diagram: Metadata

Figure 228. Metadata

3.9.2 Class Definition <<Value>>

An informal or natural language definition of a something and potentially a reference to external definitions.
A Definition may be in the context of a natural language to scope the language it is expressed in.

[ISO 1087] definition: representation of a concept (3.2.1) by a descriptive statement which serves to differentiate it from related concepts

[FUML] Comment (where body corresponds with "text definition").

Direct Supertypes

Metadata

Attributes

text definition : String Value
Text describing a something in natural language. The language may be indicated by a context of the definition.
[OWL] rdfs:comment

external reference : IRI Identifier
A reference to an external information resource that further defines something.
[FIBO] ReferenceDOcument

external term : Term
Specific term in an external resource that further defines something.

summary description : String Value
A short description of something.

3.9.3 Association Definition Relationship

Relationship between a thing and its definitions.

Direct Supertypes

Metadata relationship

Association Ends

is definition of : Entity [1] *Redefines*: ownes property type: Owned Property Type

Some thing described by a definition.
[FIBO] defines
[FUML]annotatedElement

defined by : Definition [*] *Redefines*: ownes property type: Owned Property Type

An informal description of something.
[FIBO] hasDefinition
[UML] comment
[FUML] ownedComment

3.9.4 Class Information Source <<Role>>

Metadata defining the origin or provenance of a set of statements in a model or data.
Note that the source could be a human, an organization, a mapping or other automated processes.

Direct Supertypes

Actual Entity

3.9.5 Class Metadata <<Value>>

Information about the source, provenance or origin of information. Metadata may be a managed entity, providing for provenance.
[NIEM] MetadataType

Direct Supertypes

Structured Value

3.9.6 Association Metadata relationship

Relationship between something and metadata about that thing; data about data.
[OWL] AnnotationAssertion

Association Ends

metadata about : Thing [*] *Redefines*: ownes property type: Owned Property Type

The subject of metadata, the entity described by the metadata.
[OWL] annotationSubject of Annotation Assertion

has metadata : Metadata [*] *Redefines*: ownes property type: Owned Property Type

Metadata associated with (data about the information concerning) the subject entity.
[OWL] AnnotationProperty, annotationValue of Annotation Assertion

3.9.7 Association Source of Information

Relation defining an entity making a statement represented within a model. E.g. the person or organization that made a statement.

[ISO 1087] source identifier: information in a terminological entry (3.8.2) which indicates the source documenting the terminological data (3.8.1)

Association Ends

made statement : Speech Act [1..*] *Redefines*: ownes property type: Owned Property Type

Metadata representing statements made by an authoritative source.
Sources may be people, organizations, documents, information systems, etc.

has authoritative source : Information Source [*] *Redefines*: ownes property type: Owned Property Type

Metadata representing the authority behind a statement - who or what made a statement captured in a model.

3.9.8 Class Speech Act <<Value>>

Statements provide metadata as to the source of information - who or what said it.
This source of the information may be captured using "InformationSource" metadata about the metadata.

[ISO11404] provision that conveys information

Direct Supertypes

Metadata

Attributes

date and time : Value
Metadata representing the date and time the statement was made or modified.

version : Value
Metadata representing an identifier for a version of information.

3.10 SMIF Conceptual Model::Patterns

Patterns are templates for structures or compositions of things that may then be expressed as instances of the pattern.

3.10.1 Diagram: Patterns

Figure 229. Patterns

3.10.2 Class Computed

Attributes

<<Characteristic>>computation : Expression Node [0..1]
<computation> provides an expression that computes a value for the variable based on the expression applied to the current context..

3.10.3 Association Exclusion

Association Ends

excluded by : Pattern Variable [*] *Redefines*: ownes property type: Owned Property Type

excludes : Pattern Variable [*] *Redefines*: ownes property type: Owned Property Type

3.10.4 Class Expression Variable

An expression variable defines the value of the variable as computed by <computation>. Note that expression variables are not always able to be asserted or reversed and may therefore not provide for bi-directional mapping patterns. Any ability to assert or reverse a computation is implementation specific.

Direct Supertypes

Computed, Pattern Variable

3.10.5 Class Focus Variable

A property variable of a pattern representing the extent of the subject type within the context of the owning pattern.
The value of qualification shall be "Select".
The <has type> of the variable is asserted be the same as the subject type of the pattern.

Direct Supertypes

Type Pattern Variable

3.10.6 Class Part Variable

A pattern property variable representing a part of the subject type. Additional relations and rules may be made about the part. A type with parts is by its nature a composition.

Direct Supertypes

Type Pattern Variable

Attributes

is boundary part : Boolean Value [0..1]
True if the property is on the boundary of the pattern and connectible (may have relationships) external to the pattern. e.g. "Port"

3.10.7 Class Pattern <<Intersection>>

A pattern represents a set of assertions true about individuals or sets of individuals qualified by pattern properties. All propositions asserted or negated by a pattern (as a context) are considered "templates" where identity is not required to match.

The structure of the pattern is defined by the properties and asserted (sub) situations (including relationships) that are asserted by the pattern.

In many cases the relationships and rules defined for a pattern will reference pattern properties. These relationships will

hold for instances of the pattern where things are bound to the pattern properties.

[DTV] general situation kind: situation kind that is not an individual situation kind. A situation kind is a general situation kind if it can be exemplified by more than one Event in some possible world, even when it cannot have more than one Event in the possible world chosen to be the universe of discourse.

[UML] StructuredClassifier. Also Similarity with TemplateSignature

[OWL] May be used to represent Class Expressions

Direct Supertypes

Entity Type, Identifiable Property Owner, Lexical Context, Proposition

3.10.8 Association Pattern Bindings

Association Ends

/ : Variable Binding [*] *Redefines*: ownes property type: Owned Property Type

/ : Pattern Match [1] *Redefines*: ownes property type: Owned Property Type

3.10.9 Class Pattern Match

A pattern match provides the corespondents between a pattern and the situations it matches using variable bindings.
A pattern match implies and proves that the pattern <categorizes> the situation.
The matched pattern <states> any consequences of the matching, such as the pattern <categorizes> the pattern instance.

Direct Supertypes

Actual Entity

3.10.10 Association Pattern Matches

Association Ends

/ satisfies : Pattern [1] *Redefines*: ownes property type: Owned Property Type

Pattern that is satisfied by a "Pattern Match" based on a set of "Variable Bindings".

/ satisfied by : Pattern Match [*] *Redefines*: ownes property type: Owned Property Type

Pattern match that satisfies a pattern.

3.10.11 Class Pattern of Type

A pattern of type defines a set of properties and relationships that must hold true for all instances of a type. Where the pattern includes parts, the subject type is a composition.
Patterns augment the semantics of th subject type in the context of the pattern.

Direct Supertypes

Pattern

3.10.12 Class Pattern Variable

A pattern variable is a property of a pattern that provides a contextual property within that pattern for rules and relationships to be bound to.
A pattern variable is a placeholder for all or a subset of the instances of the variables type.
Properties of an association or relationship may be bound to a pattern variable where the type of the pattern variable is

compatible with the type of the relationship's property type.

[UML] Similarity with TemplateParameter
[CL] Functional Term

Direct Supertypes

Conditional, Owned Property Type

Attributes

qualification : Variable Qualification [1]
<qualification> defines the behavior of an element with respect to a pattern - how the variable impacts the selection, evaluation or assertion of the pattern.

explicit : Boolean Value
If true, Element must be explicitly asserted as the indicted type, not derived or inferred from a supertype or super property.

3.10.13 Association Pattern Variables

Relationship defining variable properties within a pattern.

Direct Supertypes

Statement

Association Ends

owns variable : Pattern Variable [*] *Redefines*: ownes property type: Owned Property Type

A variable property defined within the context of a pattern that is used as part of the patterns definition. [UML] ownedAttribute

has owning pattern : Pattern [1] *Redefines*: ownes property type: Owned Property Type

Pattern owning a pattern variable.

3.10.14 Class Proposition Variable

A proposition variable utilizes some proposition (e.g. relationships) as a part of the definition of a pattern, it extends a basic proposition in that it adds properties to determine the effect the assertion has on pattern instances.
A Proposition Variable is a lexical scope context that <asserts> or <negates> other propositions qualified by <has strength> and <explicit>. As a lexical scope it may "own" the asserted propositions.
Proposition Variable is often used with associations and relationships to define the way pattern properties are related to other pattern properties or actual entities.
For a pattern associations, [UML] Connector. (type = has type). Each ConnectorEnd corresponds with a Structured Property Binding.

Direct Supertypes

Pattern Variable

3.10.15 Association Qualified Proposition

Association defining exactly one proposition (such as an association) qualified by a qualified proposition variable.

Association Ends

qualifies : Proposition [1] *Redefines*: ownes property type: Owned Property Type

/ qualified within : Proposition Variable [0..1] *Redefines*: ownes property type: Owned Property Type

3.10.16 Association Subject of Pattern Relationship

Relationship defining the subject pattern of a type specific pattern.

Direct Supertypes

Assertion

Association Ends

/ asserts pattern : Pattern of Type [0..*] *Redefines*: ownes property type: Owned Property Type

A pattern asserted for all instances of a type. Where the pattern includes parts, the type defines a composition.

/ subject type : Type [1] *Redefines*: ownes property type: Owned Property Type

The type which is the context of a pattern of type. The pattern is "about" the subject type.

3.10.17 Association Subsetting

In a pattern or mapping rule, defines a variable that represents a subset of another property (or if multiple, their union). The subset may be constrained by a more specific type, expressions or required cardinalities.
Subset: Set A is a subset of set B if all of the elements (if any) of set A are contained in set B

Association Ends

/ subsets : Pattern Variable [*] *Redefines*: ownes property type: Owned Property Type

Variable that a subset variable subsets. The subset variable shall be populated by a subset of the <subsets> variable based on the type and constraints of the subset variable.

/ has subset : Pattern Variable [*] *Redefines*: ownes property type: Owned Property Type

Subsets of the variable.

3.10.18 Class Type Pattern Variable

Type Pattern variable is an abstract supertype that provides for a restriction that parts and focus properties must be owned by a pattern of a type.

Direct Supertypes

Pattern Variable

3.10.19 Class Variable Binding

A variable binding defines a value for a particular variable of a particular owning pattern as part of a pattern match.

Direct Supertypes

Owned Property Binding

3.10.191 Enumeration Variable Qualification

Variable qualification values define the behavior of an element with respect to a pattern - how it impacts the selection, evaluation or assertion of the pattern.
package SMIF Conceptual Model::Patterns
public enum Variable Qualification
{Select, Optional, Assert, Negate, Exactly One, There Exists, All}

Literals

- Select

Select is used in query and mapping patterns, all elements of the classified type that match the pattern are selected as instances of the pattern.
Select may be considered a qualified "All". Select does not assert the existence of something, it determines the existence of a pattern match such that other assertions may be made.
Where a pattern is asserted, "Select" variables shall be asserted.
Relationships between properties with <quantifier>=Select must hold between the selected properties for the pattern to be asserted.

- Optional

Optional is used in query and mapping patterns, the property shall be populated as a consequence of the pattern matching.
Where a pattern is asserted, "Optional" variables shall not be asserted.
Optional is the default if no qualification is stated.

- Assert

The property does not impact the selection of the pattern, it is an asserted consequence of the pattern.

- Negate

The property does not impact the selection of the pattern, it is negated consequence of the pattern - it may not exist.

- Exactly One

The existential quantifier limited to exactly one of a potentially larger set of the properties type.

- There Exists

The existential quantifier - at least one of the properties type.

- All

The universal quantifier - the quantified property is a stand-in for all elements of the existent of the quantified type

Figure 230. Patterns

3.11 SMIF Conceptual Model::Properties

Properties define the most granular connections between entities or values. Properties may be used as the ends of relationships, to represent individual characteristics or as elements of a data structure.

3.11.1 Diagram: Characteristics

Figure 231. Characteristics

3.11.2 Diagram: Properties

Figure 232. Properties

3.11.3 Diagram: Property Foundation

Figure 233. Property Foundation

3.11.4 Class Abstract Property Owner

Direct Supertypes

Lexical Scope

3.11.5 Class Annotation Property

An annotation property is a specialization of property where the referenced elements represent metadata about the related proposition, structure or information (or model element) rather than a fact or condition of the domain being represented. For an annotation property, <is of type> describes instances of the structured type for which the property is defined. Typical uses of annotations include provenance of information, when a record was created, etc.
[ISO11404] annotation: descriptive information unit attached to a datatype, or a component of a datatype, or a procedure (value), to characterize some aspect of the representations, variables, or operations associated with values of the datatype

Direct Supertypes

Characteristic Type

3.11.6 Association Bound Individual

Relationship defining the thing bound to a subject based on a bound property - the "object" of the property binding.

Association Ends

↗ binds : Thing [1] *Redefines*: ownes property type: Owned Property Type

The thing bound to a property in a specific situation. E.g. if the weight of truck-XYZ is 4500 LBS, the bound individual would be "4500 LBS".
[FUML] value
[OWL] rdf:object

↗ bound in : Property Binding [*] *Redefines*: ownes property type: Owned Property Type

Bindings in which a thing participates.

3.11.7 Association Bound Property

Relationship defining the property type that defines the semantics of a property binding. E.g. if the weight of truck-XYZ is 4500 LBS, the bound property could be "has weight".

Direct Supertypes

Extent of Type

Association Ends

↗ categorizes binding : Property Binding [*] *Redefines*: ownes property type: Owned Property Type

Bindings referencing a property.

↗ bound by : Concrete Property Type [1] *Redefines*: ownes property type: Owned Property Type

The property a binding binds a thing to.
[FUML] definingFeature
[OWL] rdf:predicate

3.11.8 Association Bound Subject

Relationship defining the subject of a bound property. Where the subject is a relationship, the relationship becomes transparent and the applicable subject(s) are the other ends of the relationship. E.g. if the weight of truck-XYZ is 4500 LBS, the bound subject would be Truck-XYZ". Also known as "Inherence" (UFO).

Association Ends

↗ has binding : Property Binding [*] *Redefines*: ownes property type: Owned Property Type

Bindings asserted for properties, the property values inhering in the thing.

↗ bound to : Thing [1] *Redefines*: ownes property type: Owned Property Type

The subject of a property binding.
[FUML] owningInstance (note that in SMIF the owner and subject may not be the same). Where the are the same, the semantics are the same as FUML.
[OWL] rdf:subject
[UFO] Inheres in

3.11.9 Class Characteristic Binding

A characteristic of a specific thing, e.g. the color of Pump-1234 in the <bound to> entity. A characteristic is a "first class" element and may participate in relationships and have annotations.

[IDEAS] measureOfIndividual: A typeInstance that asserts an Individual is an instance of a Measure - i.e. the Individual "has" a property corresponding to the Measure.

[ISO 1087] characteristic: abstraction of a property of an object (3.1.1) or of a set of objects

[Guizzardi] Intrinsic Trope (UFO)

[DOLCE] Quality

Direct Supertypes

Actual Entity, Property Binding

3.11.10 Class Characteristic Type

A kind of characteristic a type of thing may have, e.g. paint may have a color. Characteristic type is the type of characteristic bindings which are "first class" elements and may participate in relationships and have other characteristics.

[IDEAS] Property: An IndividualType whose members all exhibit a common trait or feature. Often the Individuals are states having a property (the state of being 18 degrees centigrade), where this property can be a CategoricalProperty (qv.) or a DispositionalProperty (qv.).

[ISO 1087] type of characteristics: category of characteristics (3.2.4) which serves as the criterion of subdivision when establishing concept systems. NOTE The type of characteristics colour embraces characteristics (3.2.4) being red, blue, green, etc. The type of characteristics material embraces characteristics made of wood, metal, etc.

[FIBO] Simple Property: Simple Properties are assertions about things in a class, which may be framed in terms of some simple type of information.

[Guizzardi] Intrinsic Trope Type

[DOLCE] Quality Type

[OWL] rdf:Statement

[UML] Property

Direct Supertypes

Concrete Property Type, Entity Type

3.11.11 Class Computed Property Type

A property derived by an expression.

Direct Supertypes

Derived Property Type, Expression Context

3.11.12 Class Concrete Property Type

A concrete property type is a property type which has instances defined by property bindings.

Direct Supertypes

Property Type

3.11.13 Class Derived Property Type

A derived property type is a property type that is derived in some way from one or other properties in the same context.

Direct Supertypes

Property Type

3.11.14 Class Identifiable Property Owner

Property Owner is an abstract element for anything that may own a set of property bindings. This element is abstract and not intended to directly represent domain concepts. Subtypes of property owner provide semantic interpretation.

Direct Supertypes

Abstract Property Owner, Lexical Context

3.11.15 Class Owned Property Binding

An owned property binding defines a value for a particular property of a particular owning property type (or structure). Similar to an OWL triple, an owned property binding does not have independent identity.
Constraint: Each owned property binding must be <bound by> an owned property type that is owned by the <has type> owned type of the <bound to> property owner.

Owned property type is abstract and not intended to directly represent semantic elements.

Direct Supertypes

Property Binding

3.11.16 Class Owned Property Type

An owned property type is a property definition defined as a composite part of an association type - most often used in data structures and relationships. Association property types are the types of association property bindings. Also known as "association end".

[FIBO] Relationship Property
[FUML] memberEnd (of association) Property

Direct Supertypes

Concrete Property Type

Associations

: Binary Association Type [1] *Subsets*: property type owned by:Property Owner Type

3.11.17 Class Property Binding

A property value binding binds a particular thing (the value) to a situation based on a defined property.

Where <binds> is an expression evaluation, the property value shall evaluate to the evaluation of the expression.

Where <binds> is a property, the property value shall be the property values bound to that property in <bound to> situation.

The bound to thing must conform with the <is of type> type of the property. If the bound individual conforms to the "requires type" of the property, the <is of type> of the bound thing will be asserted.

The type of the <bound to> structure must (directly or indirectly) have the type the <bound by> properties <property of> type.

[Guizzardi] Trope (UFO)

[FUML] Slot (Noting that in SMIF the binding may or may not be owned by the subject, depending on the subtype of property).

[CL] Binding:

[OWL] Union(ObjectPropertyAssertion, DataPropertyAssertion, AnnotationAssertion), RDF Triple
=Note: RDF Triples do not have identity where as some subtypes of SMIF:Property Type do have identity and are therefor statements.

Direct Supertypes

Thing

Associations

/ <<Restriction>> : Property Type [1..*] *Redefines*: has type:Type

3.11.18 Association Property Domain

Relationship defining the set of properties defined for a type.
Where the <property of> type is a relationship type, the "subject" of the property is the other ends (properties) of the relationship.
Where the <property of> type is not a relationship, the subject of the property is the <property of> type.

Association Ends

/ has property : Property Type [*] *Redefines*: has type: Type

A property of a structured type such that there may be bindings of a thing to instances of the structured type with reference to the property which defines the semantics of the bound thing withing the context of the structure.
[FUML] feature
[UML] memberEnd. attribute (of classifier).

/ has domain : Type [0..1] *Redefines*: has type: Type

Type for which a property is relevant. The domain of the property.
<property of> excludes "Owned Property Type" and ("Association Type" that is not "Relationship Type")
[FUML] featuringClassifier
[OWL] Domain

3.11.19 Class Property Owner Type

A type of Property Owner (See Property Owner for details) which defines a set of "Owned Property Types" which are the types of owned property bindings.
Property owner is abstract and not intended to directly represent semantic elements.

Direct Supertypes

Type

3.11.20 Association Property Range

Property range is a relationship defining the possible ranges of a property. Property range may be restricted to hold within a relationship or a domain type.

Association Ends

/ has range : Type [0..1] *Redefines*: has type: Type

The range of a property reflects the type of possible values of that property type.

/ range of : Property Type [*] *Redefines*: has type: Type

Properties having a type.

3.11.21 Class Property Restriction

A property restriction is a derived property that utilizes the subset (hasSupertype) redefined properties to make additional assertions about a property in the context of its domain, which is part of its context. The property domain and other constraints apply when the domain is satisfied.
Property restrictions may also be used do define multiple domains and ranges, per OWL.

Direct Supertypes

Derived Property Type

3.11.22 Class Property Type

A property type defines the way in which instances of a type participate in (or, are involved in) instances of another type (including relationships). Sometimes called a variable, argument or role.
Owned properties are the basis for the ends of associations and relationships.
In a conceptual model the terms associated with a property kind are typically "verb phrases" defining how instances of the involved type participate in the situation or relationship.
In a value (static data structure) the property is a "slot" of a record and may have a term which is a noun or verb phrase.
So that constraints of a type flow to relationships involving that type: All propositions that hold within a type referenced by <is of type> hold within the structured type referenced by <property of>. I.e. the structured type is in the context of the types of its properties.
In a function, a property is a function argument.

[Guizzardi] Trope Type

[FUML] Parameter where owner is operation. Otherwise Property.

[UML] Property. All typed elements in SMIF are Property Types.

[CL] Operator: distinguished syntactic role played by a specified component within a functional term

[OWL] rdf:Property, ObjectUnionOf(owl:ObjectProperty, oe;DatatypeProperty).

Direct Supertypes

Type

Attributes

<<Characteristic>>is sufficient : Boolean Value
One of the set of sufficient conditions that will infer the type designated in <has domain>.
Asserts that when a property binding is <bound> to a thing, that thing will be asserted to be of the defined domain type and all other sufficient contritions are met. Sufficient conditions may be specified in properties an sufficient generalizations.
If this property is not set, assertion is based on the contextual default, such as the rules of the underlying logic. In description logics such as OWL, the default is TRUE for concrete properties and FALSE for derived properties.
If asserts domain is false it is an error to create a property binding that does not already have the domain type, the domain is a constraint.

Associations

<<Restriction>> : Property Binding [0..*] *Redefines*: categorizes:Thing

3.11.23 Association Property Type Ownership

Direct Supertypes

Statement

Association Ends

- ownes property type : Owned Property Type [*] *Redefines*: categorizes: Thing
- property type owned by : Property Owner Type [1] *Redefines*: categorizes: Thing

3.11.24 Association Proxy Delegate

A relationship defining the proxies for an owned property type, the restrictions that are contextually added to a concrete property based on the domain of the property restriction.

Association Ends

proxy of : Property Type [1] *Redefines*: categorizes: Thing

The property type that is proxied.

has proxy : Proxy Property Type [*] *Redefines*: categorizes: Thing

A type of proxy for a type of property.

3.11.25 Class Proxy Property Type

A proxy property is a derived property and a property restriction that utilizes the <proxy of> property to provide another way to reference, name or constrain that property, in general from another domain. The proxies property must be <stated by> a common context such that the traversal through that property is unambiguous.
By convention, proxy properties are verb phrases where as non-proxy owned properties are noun phrases (role names).

Direct Supertypes

Property Restriction

3.11.26 Association Redefinition

Defines the generalization as a redefinition, subsuming the more general type in the definitional context.

Where <redefines> is true the more specific type subsumes the more general type in the definition context. In this case the more general and more specific sets are equivalent. A type may be redefined multiple times, as long as it is unambiguous which definition applies for a particular instance.

Where <redefines> is false or not defined the more specific type represents a subset of the more general property.

Redefinition is most often used with properties (as defined in UML) but may also be applied to other types.

Direct Supertypes

Generalization

Association Ends

redefines : Property Type [*] *Redefines*: categorizes: Thing

redefined by : Property Type [*] *Redefines*: categorizes: Thing

3.12 SMIF Conceptual Model::Relationships

Relationships are primitive but identifiable conditions that relate other entities through properties of the relationships. Relationships have their semantics described by a relationship type. The ends of relationships are defined by "structured property type", a relationship may have any number of "ends".
Relationships are first-class "actual" and "temporal" things that exist in their own right. These are known as "external relations" in much of the theoretical literature.

3.12.1 Diagram: Relationships

Figure 234. Relationships

Relations are atomic actual situations that bind 2 or more properties as a fact.

3.12.2 Class Relationship

A relationship defines a situation involving related things. A relationship may be asserted within a context as true or false within that context. Each relationship type has a number of bindings of which do not change for the life of the relationship..
A relationship may be true or false within its context (including a timeframe) but is atomic in its truth value.
Relationships may participate in (be bound to) other relationships and as such bindings involving a relationship may change over time. That is, relationships are "first class" objects.

[IDEAS] tuple: A relationship between two or more things.
Note: SMIF allows one end of a relationship.

[OWL] An OWL class that is a subclass of SMIF: Relationship

Direct Supertypes

Actual Entity, Association

3.12.3 Class Relationship Type

A relationship type defines a type of condition, the relationship, involving related things. A relationship may be asserted within a context as true or false within that context. Each relationship type has a number of <has property> "structured property type" properties which describe the role of the related things with respect to the relationship, values of which

uniquely do not change for the life of the relationship.
A relationship may be true or false within its context (including a timeframe) but is atomic in its truth value.
Relationships may participate in (be bound to) other relationships and as such bindings involving a relationship may change over time.
The terms for properties of a relationship in a conceptual model are typically verb phrases, connecting the relationship with the related types.

[FIBO] A kind of Mediating Thing

[IDEAS] TupleType: The Powertype of tuple.

[FUML] Association where memberEnd corresponds with <has property>. Note that SMIF relationships are "first class" and may also be considered to correspond to an association class where there are any properties or other relationships referencing the subject relationship.

[UML] AssociationClass (note that "end ownership" is meaningless in SMIF).

[Guizzardi2015] Relator: endurants of a special kind, with the power of connecting (mediating) other endurants. Note: Guissardi "mediation" corresponds with relationship properties.

Direct Supertypes

Association Type, Entity Type

3.13 SMIF Conceptual Model::Top level

The top level objects provide the foundation for all objects in a SMIF model

3.13.1 Diagram: Context

Figure 235. Context

3.13.2 Diagram: Top Level

Figure 236. Top Level

Diagram showing summary of top level classes and significant subtypes.

3.13.3 Class Actual Entity

An actual entity is an identifiable, individual person, specific object, process enactment, agreement, etc. Actual Entities do not have to be physical, e.i. may denote social constructs. Actual entities are disjoint from types.
A more specific class of actual entity (e.g., Person) is intended to refine the classification of the individual thing.

Individuality (or selfhood) is the state or quality of being an individual; particularly of being separate from other individuals and possessing identity. Actual entities typically have a lifetime and some individuals may change over that lifetime. Individuals may have parts that together help define the individual but may change over time.
"Actual" does not imply current existence.

[ISO 1087] individual concept: concept (3.2.1) which corresponds to only one object

[UML] Loose correspondence with "InstanceSpecification". SMIF instances are direct instances of their types, there is no "indirection" through value specification as their is in UML.

[Guizzardi] (individual concept)

[CL] Individual: one element of the universe of discourse

[DOLCE] Particular: particulars are entities which have no instances

[SOWA1999] Independent. Can be considered "Actuality" when including social constructs in [SOWA1999] Physical.

[OWL] Individual

Figure 237. Actual Entity Detail

Direct Supertypes

Entity

3.13.4 Association Assertion

An assertion relationship between a context and the propositions asserted within that context. The <asserts> proposition is asserted (defined as "true") for all things contextualized by the <holds within> context. Assertion of truth is not absolute, it is relative to the context. For example, something could be asserted within a context where that entire context is asserted to be false.
Assertion is transitive.
[CL] Implication
[OWL] Assertion; Any [OWL] Assertion included in a graph (All assertions in an OWL graph are asserted by the graph)

Association Ends

asserts : Proposition [*] *Redefines*: categorizes: Thing

Proposition that is asserted (must be true) for anything contextualized by a context. As types are a context, types may assert a proposition for their instances.

holds within : Context [*] *Redefines*: categorizes: Thing

Context in which a proposition is asserted (required to be true). Anything contextualized by the context is subject to the proposition.

3.13.5 Class Context

A <Context> is proposition that impacts the interpretation of other propositions.

Subtypes of <Context>, such as <Type> ascribe more semantics to the context as well as limit the things it contextualizes.

[CL] Sort: any subset of the universe of discourse over which some quantifier is allowed to range

[ISO 1087] concept field: unstructured set of thematically related concepts (3.2.1)

[SOWA1999] Mediating

Figure 238. Context Detail

Direct Supertypes

Proposition

3.13.6 Class Entity

An identifiable entity is any identifiable thing other than values, this includes individuals, types, axioms, situations, speech acts, information structures, etc.
Identifiable entities always have some kind of identity and may have identifiers. Note that identity is an abstraction that may have representation in models as any number of identifiers, also known as a "sign".

[OWL] Entity type (Implied in section [OWL] 5.8) as an instance of rdfs:Class

Direct Supertypes

Thing

3.13.7 Association Extent of Context

Association Ends

contextualizes : Thing [*] *Redefines*: categorizes: Thing

in context of : Scoped Context [*] *Redefines*: categorizes: Thing

3.13.8 Class Fact

A fact is a proposition that is not qualified by a context - asserted or negated in any context. Facts are not expected to represent "truth", but a commitment accepted by an interpreter of a model.
Propositions that are not facts are, directly or indirectly, qualified by some fact.
Facts that do not assert or negate other propositions are considered "atomic".

Example: Jill believes Johnny fell in a well.
"Johnny fell in a well" is a proposition (probably a relationship).
What Jill believes is a statement of fact and also a relationship that subtypes "Assertion".

Direct Supertypes

Proposition

Associations

: Context [0] *Redefines*: holds within:Context

: Context [0] *Redefines*: negated within:Context

3.13.9 Association Negation

An assertion relationship between a context and the propositions negated (FALSE) within that context. The <negates> proposition is asserted as FALSE for all things contextualized by the <negated within> context. Assertion or negation of truth is not absolute, it is relative to the context.
[CL] Negation+Implication

Association Ends

negates : Proposition [*] *Redefines*: negated within: Context

Proposition that is negatively asserted (must be FALSE) for anything contextualized by a context.
As types are a context, types may assert or negate a proposition for their instances.

negated within : Context [*] *Redefines*: negated within: Context

Context in which a proposition is negated (required to be FALSE). Anything contextualized by the context is subject to the proposition.

3.13.10 Class Proposition

A proposition is statement, or condition with a truth value (true or false) that can be determined or asserted with some level of confidence (assessment of confidence being outside of this specification).
All "facts", statements, speech acts, relationships and rules are propositions.
Propositions may be asserted to be true within a context which they <holds within>.
For a situation, the proposition is true if the situation is actual (i.e., takes place, obtains).

[SBVR] the state of affairs is posited by the proposition and if the state of affairs were actual, the proposition would be true
[CL] Sentence: unit of logical text which is true or false, i.e. which is assigned a truth-value in an interpretation
[SOWA1999] Proposition

Figure 239. Proposition Detail

Direct Supertypes

Entity

3.13.11 Class Scoped Context

A <Scoped Context> is a context that <asserts> or <negates> propositions that hold for all things the context <contextualizes>, thus providing the link between an assertion and the set of things asserted. Likewise a context <negates> propositions that are false within the context.
Context that are not scoped apply too all things.

Direct Supertypes

Context

3.13.12 Class Thing <<Anything>>

Any thing or value that does or may exist in any possible world. Thing is the supertype of all types and may therefore participate in unbounded relations.
Instances of Thing are referred to as "a thing" in this model.

[IDEAS] Thing
[OWL] Thing
[ISO 1087] object: anything perceivable or conceivable
[FIBO] Thing
[Guizzardi] Thing
[FUML] Element
[SOWA1999] "T"
[OWL] rdfs:Resource

Figure 240. Thing Detail

3.14 SMIF Conceptual Model::Types

Types provide for ways to categorize anything based on what it is, the roles it plays or the phases it may be in. Something may be categorized by any number of types (multiple classification assumption).

3.14.1 Diagram: Type-instance

Figure 241. Type-instance

3.14.2 Diagram: Types

Figure 242. Types

3.14.3 Diagram: Types as Context

Figure 243. Types as Context

3.14.4 Class Entity Type

A type of an identifiable entity. All concrete entity instances must have at least one entity type. Entity type may be mixed with other types to fully define an entity.

[FUML] Classifier

[Guarino1994] Substantial or Pseudo-Sortal (Substantial being concrete)
[Guizzardi] A Rigid Universal.
(Rigid Universal): A universal G is rigid (or modally constant) iff for any w,w' ∈ W 3. extw(G) = extw' (G) Putting definitions 4.1 and 4.3 together, we have that for any rigid universal G the following is true 4. ext(G) = extw(G), for all w ∈ W A rigid universal is one that applies to its instances necessarily, i.e., in every possible world. Every substance sortal G is a rigid universal.
[OWL] rdfs:Class (as Entity Type does not include values). However, non=primitive values are typically represented as rdfs:Class

Direct Supertypes

Type

3.14.5 Association Extent of Type

The relation between a type and the things that type categorizes, the instances which defines the extent of the type, a set.

[IDEAS] typeInstance: A couple that asserts that a Thing is a member of a Type.

[Guizzardi] (Extension functions): Let W be a non-empty set of possible worlds and let w ∈ W be a specific world. The extension function extw(G) maps a universal G to the set of its instances in world w. The extension function ext(G) provides a mapping to the set of instances of the universal G that exist in all possible worlds, such that ext(G) = U w∈W w ext (G)

[OWL] ClassAssertion

Direct Supertypes

Extent of Context

Association Ends

/ categorizes : Thing [*] *Redefines*: negated within: Context

The set of things described by a type, the "extent" of the type.
The thing a type <categorizes> is subject to the <has assertion> propositions of the type.
[FIBO] classifies

/ has type : Type [1..*] *Redefines*: negated within: Context

A type that holds for something.
Things may have multiple types and these types may change over time.
The <categorized> thing satisfies the constraints of the <has type> type.
[FIBO] isClassifiedBy
[OWL] rdf:type

3.14.6 Association Generalization

A Type Generalization is a taxonomic relationship between a more general <has supertype> type and a more specific <has subtype> type. Each instance of the specific type is also an instance of the general type.

The extent (<categorizes> property) of the specific type is the same as or a subset of the extent of the more general type. Therefore, any statement that is true for all members of a supertype must also be true for all members of any subtype.

Note that "multiple inheritance" is supported.

[IDEAS] superSubtype: A couple relating two Types which asserts that one type is a subset of the other.

[ISO 1087] generic relation: genus-species relation relation between two concepts (3.2.1) where the intension (3.2.9) of one of the concepts includes that of the other concept and at least one additional delimiting characteristic (3.2.7)

[FIBO] Inheritance

[UML] Generalization

[Guizzardi] (Specialization relation): Let F and G be two universals such that F is a specialization of G. Then, for all w ∈ W we have that extw(F) ⊆ extw(G)

[OWL] Union(SubClassOf, SubPropertyOf)

Association Ends

/ has supertype : Type [*] *Redefines*: negated within: Context

The extent (categorizes) of the subtype is a subset of the extent of the supertype. All constraints of a supertype must also be true for any subtype

/ has subtype : Type [*] *Redefines*: negated within: Context

The more general type of a generalization. The extent of the subtype is a subset of the extent of the supertype.

3.14.7 Class Intersection Type

An intersection is a type that has an extent which is the complete intersection of the extents of all supertypes. Intersection is a stronger statement than a subtype as a subtype may not be a complete intersection.

[MathWorld] The intersection of two sets A and B is the set of elements common to A and B. This is written A intersection B, and is pronounced "A intersection B" or "A cap B."

Direct Supertypes

Type

3.14.8 Association Sufficient Generalization

One of the set of sufficient conditions that will infer the type designated in <justifies subtype>.
Asserts that if an instance is of the <sufficient supertype> and all other sufficient conditions are met, the instance may be inferred to be the <justifies subtype>.

Association Ends

/ sufficient supertype : Type [*] *Redefines*: negated within: Context

/ justifies subtype : Type [*] *Redefines*: negated within: Context

3.14.9 Class Type

A <Type> is a categorization of any thing based on specific criteria. The specific criteria may or may not be formalized in a model.
A <Type> <categorizes> a set of <Thing>s which comprises the "extent" of the type.
A <Type> is a <Context> where the things it <categorizes> are <in the context> of the <Type>.

[ISO 1087] general concept: concept (3.2.1) which corresponds to two or more objects (3.1.1) which form a group by reason of common properties
[FIBO] Classifier: a standardized classification or delineation for something, per some scheme for such delineation, within a specified context
[FUML] Type
[CL] Type:: logical framework in which expressions in the logic are classified into syntactic or lexical categories (types) and restricted to apply only to arguments of a fixed type
[Guarino1994] Discriminating Predicate
[OWL] Union(rdfs:Class, rdfs:Datatype)

Similar to:
[IDEAS] Type: A set (or class) of Things. (Note that SMIF defines the extent of a type a a set, the type is the intent of that set which forms a predicate for which all members of the set is true)

Direct Supertypes

Context, Lexical Context, Scoped Context

Associations

/ sufficient supertype : Context [0] *Redefines*: negated within:Context

through association: Sufficient Generalization

/ justifies subtype : Context [0] *Redefines*: negated within:Context

through association: Sufficient Generalization

3.14.10 Class Union Type

A Union is a type that has an extent which is the complete union of the extents of all types that specialize the Union.

[FIBO] Logical Unions

[MathWorld] Given two sets A and B, the union is the set that contains elements or objects that belong to either A or to B or to both. We write A È B

[OWL] ObjectUnionOf(ObjectUnionOf, DataUnionOf)

Direct Supertypes

Type

3.15 SMIF Conceptual Model::Values

The values package defines the concepts of values and quantities expressed in units.

Values may be differentiated from entities in that values have no independent lifetime or "identity" other than the value its self. E.g. the number 5 "just is" and can't be changed. Properties and relations referencing values can, of course, change but the values are constant.
The failure to properly express units in data models often results in errors, inefficiencies and risk. Translation and federations between models, schema and data sources that is not cognizant of the units used would be even more error prone and risky. For example, what does “Speed limit 50” mean? For these reasons the SMIF language provides specific support for specifying quantity kinds and unit types in conceptual, logical and physical models. The SMIF mapping rules may then perform the appropriate unit conversions.
The foundation of information specification in SMIF at all levels is the type system. Types specified for all properties and relations involving values must match the types of the related values. The concepts of units and values as defined in "VIM" [JCGM 200-2008] is used as the basis for defining the types used in SMIF to guarantee type safety of quantities across different representations. Since many existing models and schema do not include well defined units some effort may be required to find and then specify the implicit units based on documentation, SME interviews or inspection of data or source code. It is recommended that the units used by external models and schema be determined prior to attempting federation and integration of information based on those models or schema.

VIM [JCGM 200-2008] concepts of quantities and units

VIM defines

- quantity: property of a phenomenon, body, or substance, where the property has a magnitude that can be expressed as a number and a reference [ed. to a unit]
- kind of quantity (kind): aspect common to mutually comparable quantities
- measurement unit (unit): real scalar quantity, defined and adopted by convention, with which any other quantity of the same kind can be compared to express the ratio of the two quantities as a number

SMIF concepts of quantities and units

SMIF uses the VIM concepts to define "quantity values" and types to capture the quantity kind and unit. Types are defined for each Unit. The goals for this type based approach are:

- That it is clearly grounded in semantics as defined in VIM
- That a type may be used to specify the range of a property or relation involving unit based values.
- That a quantity value (e.g. 5 grams) be representable as a simple number with a type.
- That there is a clear type hierarchy starting with a representationally independent type in a conceptual model (e.g. mass) that can be further specialized to a specific unit in a logical model (e.g. grams) and further specialized to be represented by a physical data type (e.g. “double”).
- That external models and schema may have unit specifications asserted without changing the schema.
- That a quantity of an entity be able to be referenced without a specific quantity value being known (e.g. John’s weight).
- That systems of units such as [ISO-80000] or [OMG QUDV] (A part of SysML) be able to be directly referenced as the definition of a unit.

SMIF defines three types to realize the above goals: Quantity Kind, Unit Type, Base Unit Type. SMIF also defines Quantity Values, which are instances of unit types.
In VIM a quantity has a magnitude that is expressed as a number and a reference. The SMIF quantity value is the numeric value of such a quantity where the reference is specified by the “unit reference” property of the quantity value’s type. The quantity value’s type is a “Unit Type”. The Unit type has attributes for converting a unit to a base unit, a symbol and a unit reference. Based on VIM the unit reference may be “a measurement unit, a measurement procedure, a reference material, or a combination of such” and is specified with a description that contains reference information. In summary, the reference of a SMIF quantity value is determined indirectly through its unit type. A quantity value has exactly one unit type and exactly one Quantity Kind. A quantity value expressed in any unit of the same quantity kind may be converted to any other unit of the same quantity kind.

This type-based sapproach allows specification of a property at the conceptual (quantity kind) logical (unit type) or physical (unit type with a numeric type) levels. Such specifications use the same type-based approach used for other aspects of the models. Given this information a SMIF implementation may correctly and reliably convert between compatible types regardless of representation. Please see the specification of the value types, attributes and relationships for more detail.

Example:

- A specification for a road segment has a property “Speed limit”.
- The type of this property in a reference conceptual model is “Speed:Quantity Kind”.
- A unit “Kilometer per Hour:Unit Type” is defined as a subtype of “Speed:Quantity Kind” with a “unit reference” of “[ISO-80000.4] Kilometer per Hour”. Note that quantity kinds and unit types would normally be defined in reference models that correspond to a “system of units”.
- Miles per hour is also defined as a subtype of Speed.
- A physical schema defines “Speed-KPH: Integer”.
- A SMIF mapping rule maps “Speed limit” to “Speed-KPH” and asserts a type of “Kilometer per Hour” on the “Speed-KPH” end.
- A data file defines a road “Route One” with a speed limit of 100:KPH-Int.
- When converted to a U.S. application this speed limit of route one can be viewed as 62:MPH-Int.

3.15.1 Diagram: Values

Figure 244. Values

3.15.2 Diagram: Values Only

Figure 245. Values Only

3.15.3 Class Base Unit Type

One unit type of a quantity kind may be marked as the base unit within a system of units. The base unit provides the basis for conversions between units of the same quantity kind. The base unit always has a ratio of one and an offset of zero.

Type of a [JCGM 200:2008] measurement unit that is adopted by convention for a base quantity

[FIBO] (type of) Base Unit: a measurement unit that is defined by a system of units to be the reference measurement unit for a base quantity

There ma be at most one base unit for a quantity kind within a system of units.

Direct Supertypes

Unit Type

3.15.4 Class Quantity kind

[JCGM 200:2008] A Quantity Kind is an aspect common to mutually comparable quantities represented by one or more units. Units with a common quantity kind may be algorithmically converted to any other unit of that quantity kind. e.g. temperature.

Quantity kinds are a supertype of unit types which are then a type of all quantity values, Quantity values are mutually comparable with all other quantity values categorized by the same quantity kind.

[FIBO] QuantityKind: a categorization type for “quantity” that characterizes quantities as being mutually comparable

[DOLCE] Quality Space

Direct Supertypes

Value Type

3.15.5 Association Referenced System of Units

Relationship between a system of units and the set of unit types defined within that system.

Association Ends

defined within system : System of Units [0..1] *Redefines*: negated within: Context

The system of units in which a unit is defined and is the basis for ratio and offset.

By default the system of units is "si":
http://www.iso.org/iso/iso_catalogue/catalogue_ics/catalogue_detail_ics.htm?csnumber=30669

unit of system : Unit Type [*] *Redefines*: negated within: Context

Unit type defined within a system of units

3.15.6 Class Scalar Quantity <<Value>>

Direct Supertypes

Unit Value

Attributes

: Number

The value of a quantity that, when multiplied by the unit defined in a subtype of quantity kind, specifies a measurement value such as 3 Meters.

3.15.7 Class Structured Value <<Value>>

A value that may have sub-elements (owned properties) defined as "structure property type".

Direct Supertypes

Abstract Property Owner, Value

3.15.8 Class Structured Value Type

A structured value type is a type of value that has parts represented as properties - also used for "data types" and forms.

Direct Supertypes

Property Owner Type, Value Type

3.15.9 Class System of Units

[JCGM 200:2008] A set of base units and derived units, together with their multiples and submultiples, defined in accordance with given rules, for a given system of quantities.

[FIBO] SystemOfUnits: a set of measurement units associated with a system of quantities, together with a set of rules that assign one measurement unit to be the base unit for each base quantity in the system of quantities and a set of rules for the derivation of other units from the base units

Direct Supertypes

Namespace

3.15.10 Class Unit Type

A Unit type is a type of a quantity value referencing a specific unit. A Unit Type a required type of a property representing a quantity.

Each quantity value has a reference as defined by the "unit reference" property of the quantity value's type.

[JCGM 200:2008] A Unit is a real scalar quantity, defined and adopted by convention, with which any other quantity of the same quantity kind can be compared to express the ratio of the two quantities as a number. e.g. Degrees Centigrade, Miles.

Each unit type represents refinement of a quantity kind using generalization and is thus substitutable for that quantity kind. Typically quantity kinds are used in conceptual models and unit types in physical or logical models.

Unit types may only subtype quantity kinds or other units.

Note that unit types are not units, but the type of quantity values expressed with respect to a common unit as defined in [JCGM 200:2008].

[IDEAS] MeasureCategory: A MeasureType whose members are recognized types of MeasureInstance.

Direct Supertypes

Value Type

Attributes

ratio : Real Value

The multiplier by which to multiple the referenced unit to convert to the base unit within a system of units.

offset : Real Value

The difference between zero in the referenced unit and zero in the base unit after the ratio is applied within a system of units.

symbol : String Value

The accepted symbol for the unit referenced by the unit type

unit reference : Definition [0..1]

The unit reference is the reference to a unit shared by all quantities values that are instances of a unit type.

[JCGM 200:2008] A reference can be a measurement unit, a measurement procedure, a reference material, or a combination of such. For magnitude of a quantity.

Typical references include ISO 8000 and OMG QUDV.

3.15.11 Class Unit Value <<Value>>

A unit value is a numeric magnitude with a unit type that may be used as the value of a quantity property as defined by [JCGM 200:2008]. The reference of the quantity is defined by the "unit reference" property of the Unit Type.

e.g. 5cm is an instance of the unit type "Centimeter"

Each unit value has exactly one UNit Type as a type.

In a physical model a quantity value must have a type that specifies its unit (e.g. "Gram"). The magnitude shall be expressed using "hasValue"

[JCGM 200:2008] A quantity is a property of a phenomenon, body, or substance, where the property has a magnitude

that can be expressed as a number and a reference.

Note: A quantity as defined here is a scalar. However, a vector or a tensor, the components of which are quantities, is also considered to be a quantity.

[IDEAS] ScaleMapping: A CoupleType whose members are all the couples linking MeasurePoints to RealNumbers. The CoupleType (i.e. the set of couples) represents the scale.

[FIBO] QuantityValue: number and measurement unit together giving magnitude of a quan-tity

[Guizzardi] (quale): A point in a n-dimensional quality domain

Direct Supertypes

Structured Value

Attributes

hasValue : Measurement Value
The value of a quantity that, when multiplied by the unit defined in a subtype of quantity kind, specifies a measurement value such as 3 Meters.
[OWL] rdf:value restricted to abstract quantity

3.15.12 Class Value

A Value is an atomic. immutable piece of information without a specific lifetime or identity independent of the value. Values include numbers, strings and other atomic "primitive" data. Values also include structured values, which are immutable.

In UML values may be defined by the name of an instance specification with a value type.

[IDEAS] Representation: A SignType where all the individual Signs are intended to signify the same Thing.

[ISO11404] The identification of members of a datatype family, subtypes of a datatype, and the resulting datatypes of datatype generators may require the syntactic designation of specific values of a datatype.

[OWL] data values

Direct Supertypes

Thing

3.15.13 Class Value Type

A type categorizing values where a value is an atomic piece of information without a specific lifetime or identity independent of that value. Values include numbers, strings and other atomic "primitive" data.

[IDEAS] RepresentationType: A Type that is the Powertype of Representation.

[FUML] DataType

[ISO11404] datatype: set of distinct values, characterized by properties of those values, and by operations on those values

[OWL] rdfs:Datatype (Note that some values are represented as OWL classes)

Direct Supertypes

Type