A Data-Driven Approach To Bus Timetable Optimisation Recommendations.

V1.0.0

Generated by Doxygen 1.9.1

1 Namespace Index 1

1.1 Packages . . . 1

2 Hierarchical Index 3

2.1 Class Hierarchy . . . 3

3 Class Index 5

3.1 Class List . . . 5

4 Namespace Documentation 9

4.1 Timetable_Optimisation_Recommendations Namespace Reference . . . 9

4.2 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface Namespace Reference . . . 9

4.2.1 Enumeration Type Documentation . . . 10

4.2.1.1 Direction . . . 10

4.2.1.2 Operators . . . 10

4.3 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses Namespace Reference . . . 11

4.4 Timetable_Optimisation_Recommendations.Pages Namespace Reference . . . 11

4.5 Timetable_Optimisation_Recommendations.Properties Namespace Reference . . . 12

4.6 Timetable_Optimisation_Recommendations.Request_Manager Namespace Reference . . . 12

4.7 Timetable_Optimisation_Recommendations.Route_Analyser Namespace Reference . . . 12

4.8 Timetable_Optimisation_Recommendations.Search_Algorithm Namespace Reference . . . 12

4.9 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search Namespace Reference 12

4.10 Timetable_Optimisation_Recommendations.Timetable_Analyser Namespace Reference . . . 13

4.11 Timetable_Optimisation_Recommendations.Timetable_Evaluator Namespace Reference . . . 13

4.12 Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator Namespace Reference . . . 14

4.13 Timetable_Optimisation_Recommendations.Timetable_Simulator Namespace Reference . . . 14

4.14 Timetable_Optimisation_Recommendations.UserControls Namespace Reference . . . 14

4.15 Timetable_Optimisation_Recommendations.Windows Namespace Reference . . . 15

5 Class Documentation 17

5.1 Timetable_Optimisation_Recommendations.AdvancedProgressReporting Class Reference . . . 17

5.1.1 Detailed Description . . . 18

5.1.2 Constructor & Destructor Documentation . . . 18

5.1.2.1 AdvancedProgressReporting() [1/3] . . . 18

5.1.2.2 AdvancedProgressReporting() [2/3] . . . 18

5.1.2.3 AdvancedProgressReporting() [3/3] . . . 18

5.1.3 Member Function Documentation . . . 19

5.1.3.1 Clear() . . . 19

5.1.3.2 Update() . . . 19

5.1.4 Property Documentation . . . 19

5.1.4.1 SubValue . . . 19

5.2 Timetable_Optimisation_Recommendations.App Class Reference . . . 19

5.2.1 Detailed Description . . . 20
5.3 Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable Class Reference . . . 20
5.3.1 Detailed Description . . . 21
5.3.2 Constructor & Destructor Documentation . . . 21
5.3.2.1 BlamedBusTimeTable() . . . 21
5.3.3 Member Function Documentation . . . 21
5.3.3.1 Clone() . . . 22
5.3.3.2 GetId() . . . 22
5.3.3.3 MatchDirection() . . . 22
5.3.3.4 ProposedSchArrivalTime() . . . 22
5.3.3.5 ProposedSchDepartureTime() . . . 23
5.3.3.6 SetSuggestedToReal() . . . 23
5.3.3.7 UpdateTimes() . . . 23
5.3.3.8 UpdateTotalWeight() . . . 23
5.3.3.9 WeakIsStopSame() [1/2] . . . 24
5.3.3.10 WeakIsStopSame() [2/2] . . . 24
5.3.4 Property Documentation . . . 24
5.3.4.1 Location . . . 25
5.3.4.2 SlackWeights . . . 25
5.4 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperatorFactory Class Reference . . . 25
5.4.1 Detailed Description . . . 25
5.4.2 Member Function Documentation . . . 26
5.4.2.1 SetOperatorAsync() . . . 26
5.4.3 Property Documentation . . . 26
5.4.3.1 Instance . . . 26
5.4.3.2 Operator . . . 26
5.5 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub Class Reference . . . 27
5.5.1 Detailed Description . . . 27
5.5.2 Member Function Documentation . . . 28
5.5.2.1 GetId() . . . 28
5.5.2.2 MatchDirection() . . . 28
5.5.2.3 WeakIsStopSame() [1/2] . . . 28
5.5.2.4 WeakIsStopSame() [2/2] . . . 29
5.6 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.CacheWriter Class Reference . . . 29
5.6.1 Detailed Description . . . 30
5.6.2 Member Function Documentation . . . 30
5.6.2.1 WriteToCache() . . . 30
5.7 Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster Class Reference . . . 30
5.7.1 Detailed Description . . . 31

5.7.2 Constructor & Destructor Documentation . . . 31
5.7.2.1 Cluster() . . . 31
5.7.3 Member Function Documentation . . . 31
5.7.3.1 AddDate() . . . 31
5.7.3.2 CalculateGrouping() . . . 32
5.7.3.3 GetAssociatedService() . . . 32
5.7.4 Property Documentation . . . 32
5.7.4.1 AssociatedTimes . . . 32
5.7.4.2 BusTimeTables . . . 32
5.7.4.3 ClusterId . . . 33
5.7.4.4 GroupingAssociated . . . 33
5.8 Timetable_Optimisation_Recommendations.UserControls.ClusterCard Class Reference . . . 33
5.8.1 Detailed Description . . . 33
5.8.2 Constructor & Destructor Documentation . . . 33
5.8.2.1 ClusterCard() . . . 34
5.9 Timetable_Optimisation_Recommendations.Pages.DateSelector Class Reference . . . 34
5.9.1 Detailed Description . . . 34
5.9.2 Constructor & Destructor Documentation . . . 35
5.9.2.1 DateSelector() . . . 35
5.9.3 Property Documentation . . . 35
5.9.3.1 StartDate . . . 35
5.10 Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan Class Reference 35
5.10.1 Detailed Description . . . 36
5.10.2 Constructor & Destructor Documentation . . . 36
5.10.2.1 DateSpan() [1/2] . . . 36
5.10.2.2 DateSpan() [2/2] . . . 37
5.10.3 Member Function Documentation . . . 37
5.10.3.1 AddDate() . . . 37
5.10.3.2 IsWeekDay() . . . 37
5.10.3.3 ToString() . . . 38
5.10.3.4 TotalSpan() . . . 38
5.10.4 Property Documentation . . . 38
5.10.4.1 End . . . 38
5.10.4.2 Start . . . 38
5.11 Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeSimulator Class Reference . . . 39
5.11.1 Detailed Description . . . 39
5.11.2 Constructor & Destructor Documentation . . . 39
5.11.2.1 DwellTimeSimulator() . . . 39
5.11.3 Member Function Documentation . . . 40
5.11.3.1 ProduceEstimatedDwell() . . . 40
5.12 Timetable_Optimisation_Recommendations.Pages.Evaluator Class Reference . . . 40

5.12.1 Detailed Description . . . 41
5.12.2 Constructor & Destructor Documentation . . . 41
5.12.2.1 Evaluator() . . . 41
5.12.3 Property Documentation . . . 41
5.12.3.1 Reporter . . . 41
5.13 Timetable_Optimisation_Recommendations.Timetable_Analyser.Group Class Reference . . . 41
5.13.1 Detailed Description . . . 42
5.13.2 Constructor & Destructor Documentation . . . 42
5.13.2.1 Group() . . . 42
5.13.3 Member Function Documentation . . . 42
5.13.3.1 GetStringAsync() . . . 43
5.13.3.2 GroupingsOfClusters() . . . 43
5.13.4 Property Documentation . . . 43
5.13.4.1 Grouping . . . 43
5.14 Timetable_Optimisation_Recommendations.Windows.Highlight Class Reference . . . 44
5.14.1 Detailed Description . . . 44
5.14.2 Property Documentation . . . 44
5.14.2.1 X . . . 44
5.15 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoricTimeTable Interface Reference . . . 45
5.15.1 Detailed Description . . . 45
5.15.2 Member Function Documentation . . . 45
5.15.2.1 CouldBeSolid() . . . 45
5.15.2.2 GetSolid() . . . 46
5.16 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator Interface Reference . . . 46
5.16.1 Detailed Description . . . 47
5.16.2 Member Function Documentation . . . 47
5.16.2.1 ForceUpdateCache() . . . 47
5.16.2.2 GetLocation() . . . 47
5.16.2.3 GetService() . . . 47
5.16.2.4 GetServices() . . . 48
5.16.2.5 InvalidateCache() . . . 48
5.16.2.6 IsLocation() . . . 48
5.16.2.7 IsService() . . . 49
5.17 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService Interface Reference . . . 49
5.17.1 Detailed Description . . . 50
5.17.2 Member Function Documentation . . . 50
5.17.2.1 GetArchivedTimeTable() . . . 50
5.17.2.2 GetLocations() . . . 51
5.17.2.3 GetTimeTable() . . . 51
5.17.2.4 IsArchivedTimeTableCached() . . . 51

5.17.2.5 IsTimeTableCached() . . . 52
5.17.2.6 IsWeakServiceSame() . . . 52
5.17.3 Property Documentation . . . 53
5.17.3.1 ServiceId . . . 53
5.18 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusSolidHistoricTime↩ Table Interface Reference . . . 53
5.18.1 Detailed Description . . . 54
5.19 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop Interface Reference . . . 54
5.19.1 Detailed Description . . . 55
5.19.2 Member Function Documentation . . . 55
5.19.2.1 GetArchivedTimeTable() . . . 55
5.19.2.2 GetServices() . . . 55
5.19.2.3 GetWeakArchivedTimeTable() . . . 55
5.19.2.4 IsArchivedTimeTableCached() . . . 56
5.20 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable Interface Reference . . . 56
5.20.1 Detailed Description . . . 57
5.20.2 Member Function Documentation . . . 57
5.20.2.1 GetId() . . . 57
5.20.2.2 MatchDirection() . . . 58
5.20.2.3 WeakIsStopSame() [1/2] . . . 58
5.20.2.4 WeakIsStopSame() [2/2] . . . 58
5.20.3 Property Documentation . . . 59
5.20.3.1 IsTimingPoint . . . 59
5.20.3.2 JourneyCode . . . 59
5.20.3.3 RunningBoard . . . 59
5.20.3.4 Service . . . 60
5.21 Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime Class Reference . . . 60
5.21.1 Detailed Description . . . 60
5.21.2 Constructor & Destructor Documentation . . . 60
5.21.2.1 JourneyTime() . . . 60
5.21.3 Property Documentation . . . 61
5.21.3.1 TravelTime . . . 61
5.22 Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator Class Reference . . . 61
5.22.1 Detailed Description . . . 62
5.22.2 Constructor & Destructor Documentation . . . 62
5.22.2.1 JourneyTimeSimulator() . . . 62
5.22.3 Member Function Documentation . . . 62
5.22.3.1 ProduceEstimatedTravelTimes() . . . 62
5.23 Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord Class Reference . . . 63

5.23.1 Detailed Description . . . 63
5.23.2 Constructor & Destructor Documentation . . . 63
5.23.2.1 LatenessRecord() . . . 63
5.23.3 Property Documentation . . . 64
5.23.3.1 IsLate . . . 64
5.23.3.2 Lateness . . . 64
5.24 Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessReport Struct Reference . . . 64
5.24.1 Detailed Description . . . 65
5.24.2 Property Documentation . . . 65
5.24.2.1 Service . . . 65
5.25 Timetable_Optimisation_Recommendations.UserControls.LatenessSummary Class Reference . . . 65
5.25.1 Detailed Description . . . 65
5.26 Timetable_Optimisation_Recommendations.Pages.MainPage Class Reference . . . 66
5.26.1 Detailed Description . . . 66
5.26.2 Constructor & Destructor Documentation . . . 66
5.26.2.1 MainPage() . . . 66
5.27 Timetable_Optimisation_Recommendations.Windows.MainWindow Class Reference . . . 67
5.27.1 Detailed Description . . . 67
5.28 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move Struct Reference 67
5.28.1 Detailed Description . . . 68
5.28.2 Member Function Documentation . . . 68
5.28.2.1 ToString() . . . 68
5.28.3 Property Documentation . . . 68
5.28.3.1 Service . . . 68
5.29 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.MoveSelector Class Reference . . . 68
5.29.1 Detailed Description . . . 69
5.29.2 Constructor & Destructor Documentation . . . 69
5.29.2.1 MoveSelector() . . . 69
5.29.3 Member Function Documentation . . . 69
5.29.3.1 BestMoveSelectorAsync() . . . 69
5.30 Timetable_Optimisation_Recommendations.Pages.MovesMade Class Reference . . . 70
5.30.1 Detailed Description . . . 70
5.30.2 Constructor & Destructor Documentation . . . 70
5.30.2.1 MovesMade() . . . 70
5.31 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.NeighbourhoodSolution Class Reference . . . 71
5.31.1 Detailed Description . . . 71
5.31.2 Constructor & Destructor Documentation . . . 71
5.31.2.1 NeighbourhoodSolution() . . . 71
5.31.3 Member Function Documentation . . . 72
5.31.3.1 GenerateNegibourhood() . . . 72

5.32 Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult > Class Template Reference . . . 72
5.32.1 Detailed Description . . . 73
5.32.2 Constructor & Destructor Documentation . . . 74
5.32.2.1 NotifyTaskCompletion() . . . 74
5.32.3 Property Documentation . . . 74
5.32.3.1 ErrorMessage . . . 74
5.32.3.2 Exception . . . 74
5.32.3.3 InnerException . . . 74
5.32.3.4 IsCanceled . . . 75
5.32.3.5 IsCompleted . . . 75
5.32.3.6 IsFaulted . . . 75
5.32.3.7 IsNotCompleted . . . 75
5.32.3.8 IsSuccessfullyCompleted . . . 75
5.32.3.9 Result . . . 75
5.32.3.10 Status . . . 76
5.32.3.11 Task . . . 76
5.32.3.12 TaskCompletion . . . 76
5.32.4 Event Documentation . . . 76
5.32.4.1 PropertyChanged . . . 76
5.33 Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.Performance↩ Evaluator Class Reference . . . 76
5.33.1 Detailed Description . . . 77
5.33.2 Member Function Documentation . . . 77
5.33.2.1 AddRecords() . . . 77
5.33.2.2 GenerateLatenessReport() . . . 77
5.34 Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluatorChecks Class Reference . . . 78
5.34.1 Detailed Description . . . 78
5.34.2 Constructor & Destructor Documentation . . . 78
5.34.2.1 PreEvaluatorChecks() . . . 78
5.34.3 Member Function Documentation . . . 79
5.34.3.1 DownloadAllFilesNeeded() . . . 79
5.34.3.2 EvaluateTimeTable() . . . 79
5.35 Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload Class Reference . . . 79
5.35.1 Detailed Description . . . 80
5.35.2 Constructor & Destructor Documentation . . . 80
5.35.2.1 PreliminaryDataDownload() . . . 80
5.36 Timetable_Optimisation_Recommendations.Pages.PreviousPerformance Class Reference . . . 80
5.36.1 Detailed Description . . . 81
5.36.2 Constructor & Destructor Documentation . . . 81
5.36.2.1 PreviousPerformance() . . . 81
5.36.3 Property Documentation . . . 81

5.36.3.1 ServiceCardCollection . . . 81
5.37 Timetable_Optimisation_Recommendations.ProgressReporting Class Reference . . . 82
5.37.1 Detailed Description . . . 83
5.37.2 Constructor & Destructor Documentation . . . 83
5.37.2.1 ProgressReporting() [1/2] . . . 83
5.37.2.2 ProgressReporting() [2/2] . . . 83
5.37.3 Member Function Documentation . . . 83
5.37.3.1 Clear() . . . 83
5.37.3.2 OnPropertyChanged() . . . 84
5.37.3.3 Update() . . . 84
5.37.4 Event Documentation . . . 84
5.37.4.1 PropertyChanged . . . 84
5.38 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩ Operator Class Reference . . . 85
5.38.1 Detailed Description . . . 86
5.38.2 Member Function Documentation . . . 86
5.38.2.1 ForceUpdateCache() . . . 86
5.38.2.2 GetLocation() . . . 86
5.38.2.3 GetService() . . . 86
5.38.2.4 GetServices() . . . 87
5.38.2.5 Initialise() . . . 87
5.38.2.6 InvalidateCache() . . . 88
5.38.2.7 IsLocation() . . . 88
5.38.2.8 IsService() . . . 88
5.39 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩ Service Class Reference . . . 89
5.39.1 Detailed Description . . . 90
5.39.2 Member Function Documentation . . . 90
5.39.2.1 GetArchivedTimeTable() . . . 90
5.39.2.2 GetLocations() . . . 90
5.39.2.3 GetTimeTable() . . . 91
5.39.2.4 IsArchivedTimeTableCached() . . . 91
5.39.2.5 IsTimeTableCached() . . . 92
5.39.2.6 IsWeakServiceSame() . . . 92
5.40 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩ Stop Class Reference . . . 92
5.40.1 Detailed Description . . . 93
5.40.2 Member Function Documentation . . . 93
5.40.2.1 GetArchivedTimeTable() . . . 93
5.40.2.2 GetServices() . . . 94
5.40.2.3 GetWeakArchivedTimeTable() . . . 94
5.40.2.4 IsArchivedTimeTableCached() . . . 95

5.41 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩ Table Class Reference . . . 95
5.41.1 Detailed Description . . . 96
5.41.2 Member Function Documentation . . . 96
5.41.2.1 GetId() . . . 96
5.41.2.2 MatchDirection() . . . 96
5.41.2.3 WeakIsStopSame() [1/2] . . . 97
5.41.2.4 WeakIsStopSame() [2/2] . . . 97
5.42 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩ TableHistoric Class Reference . . . 98
5.42.1 Detailed Description . . . 98
5.42.2 Member Function Documentation . . . 98
5.42.2.1 CouldBeSolid() . . . 99
5.42.2.2 GetSolid() . . . 99
5.43 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩ TableSolidHistoric Class Reference . . . 99
5.43.1 Detailed Description . . . 100
5.43.2 Member Function Documentation . . . 100
5.43.2.1 operator RbTimeTableSolidHistoric() . . . 100
5.44 Timetable_Optimisation_Recommendations.Properties.Resources Class Reference . . . 100
5.44.1 Detailed Description . . . 101
5.44.2 Property Documentation . . . 101
5.44.2.1 AverageLateness . . . 101
5.44.2.2 Cluster_NoOfDays . . . 101
5.44.2.3 Culture . . . 101
5.44.2.4 OnTimePercentage . . . 102
5.44.2.5 ResourceManager . . . 102
5.44.2.6 ServiceString . . . 102
5.45 Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment Class Reference . . . 102
5.45.1 Detailed Description . . . 103
5.45.2 Member Function Documentation . . . 103
5.45.2.1 LengthOfSegment() . . . 103
5.45.3 Property Documentation . . . 103
5.45.3.1 SecondaryService . . . 103
5.45.3.2 Stops . . . 103
5.46 Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection Class Reference . . . 104
5.46.1 Detailed Description . . . 104
5.46.2 Member Function Documentation . . . 104
5.46.2.1 AddService() . . . 104
5.46.2.2 GetAllServicesAsync() . . . 105
5.46.2.3 GetAllSharedBusStopsAsync() . . . 105
5.46.2.4 GetServices() . . . 106

5.46.2.5 GetSharedBusStopsAsync() . . . 106
5.46.2.6 InitialiseAsync() . . . 106
5.46.2.7 RemoveServiceAsync() . . . 107
5.47 Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder Class Reference . . . 107
5.47.1 Detailed Description . . . 108
5.47.2 Constructor & Destructor Documentation . . . 108
5.47.2.1 RouteSegmentFinder() . . . 108
5.47.3 Member Function Documentation . . . 108
5.47.3.1 FindSharedRouteSegmentsAsync() . . . 108
5.47.3.2 GetServicesInSegments() . . . 109
5.47.4 Property Documentation . . . 109
5.47.4.1 PrimaryService . . . 109
5.48 Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector Class Reference . . . 109
5.48.1 Detailed Description . . . 110
5.48.2 Constructor & Destructor Documentation . . . 110
5.48.2.1 RouteSegmentSelector() . . . 110
5.49 Timetable_Optimisation_Recommendations.UserControls.ServiceCard Class Reference . . . 110
5.49.1 Detailed Description . . . 111
5.50 Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator Class Reference . . . 111
5.50.1 Detailed Description . . . 111
5.50.2 Constructor & Destructor Documentation . . . 111
5.50.2.1 ServiceCohesionEvaluator() . . . 111
5.50.3 Member Function Documentation . . . 112
5.50.3.1 FindBlameServiceCohesion() . . . 112
5.51 Timetable_Optimisation_Recommendations.Windows.Settings Class Reference . . . 112
5.51.1 Detailed Description . . . 113
5.51.2 Constructor & Destructor Documentation . . . 113
5.51.2.1 Settings() . . . 113
5.52 Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator Class Reference . . . 113
5.52.1 Detailed Description . . . 114
5.52.2 Constructor & Destructor Documentation . . . 114
5.52.2.1 SlackTimeEvaluator() . . . 114
5.52.3 Member Function Documentation . . . 114
5.52.3.1 FindBlameSlackTime() . . . 114
5.52.3.2 FindSingleBlameSlackTime() . . . 115
5.52.3.3 StandardiseSolution() . . . 115
5.53 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution Class Reference . . . 115
5.53.1 Detailed Description . . . 116
5.53.2 Constructor & Destructor Documentation . . . 116
5.53.2.1 Solution() . . . 116
5.53.3 Member Function Documentation . . . 117

5.53.3.1 CalculateTotalBlames() . . . 117
5.53.3.2 Clone() . . . 117
5.53.3.3 ObjectiveFunctionValue() . . . 117
5.53.3.4 ReplaceMove() . . . 117
5.53.3.5 ScoreOfService() . . . 118
5.53.4 Property Documentation . . . 118
5.53.4.1 BusTimeTables . . . 118
5.54 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator Class Reference . . . 119
5.54.1 Detailed Description . . . 119
5.54.2 Member Function Documentation . . . 119
5.54.2.1 ForceUpdateCache() . . . 119
5.54.2.2 GetLocation() . . . 120
5.54.2.3 GetService() . . . 120
5.54.2.4 GetServices() . . . 120
5.54.2.5 InvalidateCache() . . . 121
5.54.2.6 IsLocation() . . . 121
5.54.2.7 IsService() . . . 121
5.55 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList Class Reference . . . 122
5.55.1 Detailed Description . . . 122
5.55.2 Constructor & Destructor Documentation . . . 122
5.55.2.1 TabuList() . . . 122
5.55.3 Member Function Documentation . . . 123
5.55.3.1 FreeUpTabuListEarly() . . . 123
5.55.3.2 IsTabu() [1/2] . . . 123
5.55.3.3 IsTabu() [2/2] . . . 123
5.55.3.4 SetTabu() . . . 124
5.56 Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator Class Reference . . . 124
5.56.1 Detailed Description . . . 124
5.56.2 Member Function Documentation . . . 124
5.56.2.1 CalculateInverseWeight() [1/2] . . . 125
5.56.2.2 CalculateInverseWeight() [2/2] . . . 126
5.56.2.3 GenerateWeightedAverage() . . . 126
5.57 Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator Class Reference . . . 127
5.57.1 Detailed Description . . . 127
5.57.2 Constructor & Destructor Documentation . . . 127
5.57.2.1 TimeTableEvaluator() . . . 127
5.57.3 Member Function Documentation . . . 128
5.57.3.1 PerformIterationAsync() . . . 128
5.57.4 Member Data Documentation . . . 128
5.57.4.1 string . . . 128

5.57.5 Property Documentation . . . 128
5.57.5.1 RelatedDates . . . 129
5.58 Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTableGrouper Class Reference 129
5.58.1 Detailed Description . . . 129
5.58.2 Constructor & Destructor Documentation . . . 129
5.58.2.1 TimeTableGrouper() . . . 129
5.58.3 Member Function Documentation . . . 130
5.58.3.1 FindGroupings() . . . 130
5.59 Timetable_Optimisation_Recommendations.Request_Manager.TimetableRetrieval Class Reference 130
5.59.1 Detailed Description . . . 131
5.59.2 Member Function Documentation . . . 131
5.59.2.1 GetHistoricTimeTableBatch() [1/2] . . . 131
5.59.2.2 GetHistoricTimeTableBatch() [2/2] . . . 131
5.59.2.3 GetTimeTableBatch() . . . 132
5.60 Timetable_Optimisation_Recommendations.Windows.ViewTimetable Class Reference . . . 132
5.60.1 Detailed Description . . . 133
5.60.2 Constructor & Destructor Documentation . . . 133
5.60.2.1 ViewTimetable() . . . 133
5.61 Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted Class Reference . 133
5.61.1 Detailed Description . . . 134
5.61.2 Constructor & Destructor Documentation . . . 134
5.61.2.1 ViewTimetableHighlighted() . . . 134
5.62 Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights Class Reference . . . 134
5.62.1 Detailed Description . . . 135
5.62.2 Member Function Documentation . . . 135
5.62.2.1 Clone() . . . 135
5.62.3 Property Documentation . . . 135
5.62.3.1 Weight . . . 136

Index **137**

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Timetable_Optimisation_Recommendations 9
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface 9
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses 11
Timetable_Optimisation_Recommendations.Pages 11
Timetable_Optimisation_Recommendations.Properties 12
Timetable_Optimisation_Recommendations.Request_Manager 12
Timetable_Optimisation_Recommendations.Route_Analyser 12
Timetable_Optimisation_Recommendations.Search_Algorithm 12
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search 12
Timetable_Optimisation_Recommendations.Timetable_Analyser 13
Timetable_Optimisation_Recommendations.Timetable_Evaluator 13
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator 14
Timetable_Optimisation_Recommendations.Timetable_Simulator 14
Timetable_Optimisation_Recommendations.UserControls 14
Timetable_Optimisation_Recommendations.Windows 15

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Application
- Timetable_Optimisation_Recommendations.App . . . 19

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperatorFactory . . . 25
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.CacheWriter . . . 29
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster . . . 30
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan . . . 35
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group . . . 41
Timetable_Optimisation_Recommendations.Windows.Highlight . . . 44
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator . . . 46
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩ Operator . . . 85
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator . . . 119

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService . . . 49
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService 89

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop . . . 54
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop . 92

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable . . . 56
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub . . . 27
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoricTimeTable . . . 45
  - Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.Rb↩ TimeTableHistoric . . . 98
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusSolidHistoricTimeTable . 53
  - Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.Rb↩ TimeTableSolidHistoric . . . 99
- Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTable 95
  - Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.Rb↩ TimeTableHistoric . . . 98
  - Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.Rb↩ TimeTableSolidHistoric . . . 99
- Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable . . . 20

ICloneable
- Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution . . . 115
- Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable . . . 20
- Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights . . . 134

INotifyPropertyChanged
- Timetable_Optimisation_Recommendations.ProgressReporting . . . 82
  - Timetable_Optimisation_Recommendations.AdvancedProgressReporting . . . 17
- Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult > . . . 72

Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime . . . 60
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord . . . 63
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessReport . . . 64
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move . . . 67
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.MoveSelector . . . 68
NavigationWindow
- Timetable_Optimisation_Recommendations.Windows.MainWindow . . . 67

Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.NeighbourhoodSolution . . . 71
Page
- Timetable_Optimisation_Recommendations.Pages.DateSelector . . . 34
- Timetable_Optimisation_Recommendations.Pages.Evaluator . . . 40
- Timetable_Optimisation_Recommendations.Pages.MainPage . . . 66
- Timetable_Optimisation_Recommendations.Pages.MovesMade . . . 70
- Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload . . . 79
- Timetable_Optimisation_Recommendations.Pages.PreviousPerformance . . . 80
- Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector . . . 109

Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.PerformanceEvaluator . . . 76
Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluatorChecks . . . 78
Timetable_Optimisation_Recommendations.Properties.Resources . . . 100
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment . . . 102
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection . . . 104
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder . . . 107
Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator . . . 111
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator . . . 113
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList . . . 122
Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator . . . 124
- Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeSimulator . . . 39
- Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator . . . 61

Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator . . . 127
Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTableGrouper . . . 129
Timetable_Optimisation_Recommendations.Request_Manager.TimetableRetrieval . . . 130
UserControl
- Timetable_Optimisation_Recommendations.UserControls.ClusterCard . . . 33
- Timetable_Optimisation_Recommendations.UserControls.LatenessSummary . . . 65
- Timetable_Optimisation_Recommendations.UserControls.ServiceCard . . . 110

Window
- Timetable_Optimisation_Recommendations.Windows.Settings . . . 112
- Timetable_Optimisation_Recommendations.Windows.ViewTimetable . . . 132
- Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted . . . 133

Chapter 3

Class Index

3.1 Class List

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

Timetable_Optimisation_Recommendations.AdvancedProgressReporting
: Advanced Progress Reporting, is used to report back to the GUI on the progress of a more complex task that contains sub-tasks . . . 17

Timetable_Optimisation_Recommendations.App
: Interaction logic for App.xaml . . . 19

Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable
: A "blamed" bus timetable is a timetable record that also contains weights and blame values, from squeaky wheel optimization. Along with the logic for updating the scheduled times . . . 20

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperatorFactory
: A singleton factory object, that can produce and return back the current IBusOperator object. This allows for support of several bus operators, not just Reading Buses. You would set the operator you want to get data for and the factory will then return a singleton reference to the operator object . . . 25

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub
: Used when you need to problematically make a new IBusTimetable record, for example when you are making a new timetable during the evaluator . . . 27

Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.CacheWriter
: Used to write a cache files to the disk . . . 29

Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster
: A cluster is a selection of dates which all have the same timetable. A group is a span of consecutive days within the cluster, that all have this same timetable . . . 30

Timetable_Optimisation_Recommendations.UserControls.ClusterCard
: The card used to show a timetable cluster and the groupings within it . . . 33

Timetable_Optimisation_Recommendations.Pages.DateSelector
: The date selector page, this is where the user is asked to input a a date range, for which they intend upon making a new timetable for . . . 34

Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan
: A class used to represent a date span, which is a period between two dates . . . 35

Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeSimulator
: Used to calculate how long a bus is going to need to dwell at a bus stop, given the time of day and hence changes in passenger demand. Can be considered how busy a stop is . . . 39

Timetable_Optimisation_Recommendations.Pages.Evaluator
: The actual main evaluator stage of the program, this is the GUI for the "main part". But the page mainly contains the GUI to let the user know of the progress . . . 40

Timetable_Optimisation_Recommendations.Timetable_Analyser.Group
: A group is a collection of consecutive days within a cluster . . . 41

Timetable_Optimisation_Recommendations.Windows.Highlight
Used to tell the timetable data grid what colour highlights the cells should be . . . 44
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoricTimeTable
Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop . . . 45
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator
Provides all the information about a bus operator and gives the ability to query it further . . . 46
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService
A class which represents a single bus service . . . 49
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusSolidHistoricTimeTable
Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop. Solid, contains only records that had actual values recorded. Any non-solid record does not necessarily contain values . . . 53
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop
A class which represents a single bus stop . . . 54
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable
A class which represents a single time table record, this is a time and particular bus stop for one service . . . 56
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime
A simplistic class used to help represent the journey time between two stops . . . 60
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator
The journey time simulator class takes in a theoretical departure time, two stops to travel between and some other information. It then estimates how long it would likely take to travel between the two stops at this time of day . . . 61
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord
Used to represent the lateness of one single record . . . 63
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessReport
Used for MVVM to bind to the GUI, this contains data summerising the performance of a service 64
Timetable_Optimisation_Recommendations.UserControls.LatenessSummary
The lateness summary report card, shows the performance of the service historically . . . 65
Timetable_Optimisation_Recommendations.Pages.MainPage
The main start page for the program, this is where the user will select the primary service that they wish to optimize for . . . 66
Timetable_Optimisation_Recommendations.Windows.MainWindow
The main entry point for the application, but the logic is contained within the page . . . 67
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move
Used to represent a single move of the search algorithm, this involves making a change to one services timetable, on one running-board. A single timetable record is moved and then the surrounding records will also need to be edited in forwards and backwards propagation . . . 67
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.MoveSelector
The move-selector is used to select the best move out of an array of moves, the neighborhood, to generate a new solution . . . 68
Timetable_Optimisation_Recommendations.Pages.MovesMade
Used to display the moves that have been made by the search algorithm after it has completed and displays the finalized timetables . . . 70
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.NeighbourhoodSolution
Used to generate the neighbourhood of solutions for the search algorithm . . . 71
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >
Helper class to wrap around a Task to provide more information usable for UI databinding scenarios. As discussed in MSDN Magazine: https://msdn.microsoft.com/magazine/dn605875 . . . 72
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.PerformanceEvaluator
A class used to generate the performance of the historical/current timetables . . . 76
Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluatorChecks
Pre-evaluator checks is run before actually running the real evaluator and is used to download all of the data that is required and evaluate the performance of the old timetable . . . 78

Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload
The preliminary data download lets you download all of the data files that you are going to need for the search. It also lets you see the services past performance 79
Timetable_Optimisation_Recommendations.Pages.PreviousPerformance
Once all of the data has been downloaded in the pre-evaluator checks show the performance metrics to the user. This is the final stage before starting the actual search 80
Timetable_Optimisation_Recommendations.ProgressReporting
Used to report back the progress of th task to the GUI 82
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator
An Implementation of the IBusOperator interface for the Reading Buses API 85
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService
An Implementation of the IBusService interface for the Reading Buses API 89
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop
An Implementation of the IBusStop interface for the Reading Buses API 92
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTable
An Implementation of the IBusTimetable interface for the Reading Buses API 95
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableHistoric
An Implementation of the IBusHistoricTimetable interface for the Reading Buses API 98
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableSolidHistoric
An Implementation of the IBusSolidHistoricTimeTable interface for the Reading Buses API . . . 99
Timetable_Optimisation_Recommendations.Properties.Resources
A strongly-typed resource class, for looking up localized strings, etc 100
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment
Route Segment is set of consecutive stops that two services share, the primary service, identified in the RouteSegmenetFinder and the secondary service that also shares it 102
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection
The RouteSegmentCollection class manages the results of the RouteSegmentFinder, and provides the logic for part of the GUI, which lets the user add or remove a service of interest . . . 104
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder
The route segment finder class takes in a Bus Service (known as the primary service) and a tolerance. It then finds any service which shares a common-route segment with it, to the specified minimum segment length tolerance 107
Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector
The route-segment selector page finds other services that share a common route-segment with the primary service. And then allows the user to accept secondary services to also optimism their timetables for 109
Timetable_Optimisation_Recommendations.UserControls.ServiceCard
The service card, use to display the single service 110
Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator
The service cohesion evaluator works with the Evaluator class to calculate how well a services timetable works, with another service that shares a common route segment. By assigning a blame value to each timetable record based on how un-cohesive it is 111
Timetable_Optimisation_Recommendations.Windows.Settings
The main settings page for the application 112
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator
The slack time evaluator works with the Evaluator class to calculate if the timetable has excessive slack (Or not enough). By assigning a blame value to each timetable record based on how disruptive it is to the timetable 113
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution
Used to represent a single solution to the problem, the actual solution is stored in a dictionary, where the service is the key and an array of timetable records is the value 115
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator
The default operator if none is selected, this is mainly used to satisfy the null-ability requirement of C# then anything else 119
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList
TabuList keeps track of what moves are tabu/invalid and what moves can now be performed. Moves are tabu while the tabu tenure is greater than zero 122

Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator
Shared common code used between both the Journey Time Simulator and the Dwell Time Simulator. Manila the weighted averaging code and accuracy measures 124
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator
The main evaluator entry point, give it a set of dates where you want to use data from, an initial starting solution and a route-segment collection and then it can improve upon the timetable . . 127
Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTableGrouper
The grouper class can be used to help find patterns in the timetable where several days shared the same timetable 129
Timetable_Optimisation_Recommendations.Request_Manager.TimetableRetrieval
Used to get a batch/ multiple-days worth of timetable data in a single query. However, this still has to be done sequentially because the Reading Buses API doesn't like having more than one database connection open at once 130
Timetable_Optimisation_Recommendations.Windows.ViewTimetable
Used to display the timetable to the user 132
Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted
Interaction logic for ViewTimetable.xaml 133
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights
The weights class is used to store a target arrival and departure time, along with a raw and standardised weight for how much it should pull towards it 134

Chapter 4

Namespace Documentation

4.1 Timetable_Optimisation_Recommendations Namespace Reference

Classes

- class AdvancedProgressReporting

  Advanced Progress Reporting, is used to report back to the GUI on the progress of a more complex task that contains sub-tasks.
- class App

  Interaction logic for App.xaml
- class ProgressReporting

  Used to report back the progress of th task to the GUI.

4.2 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface Namespace Reference

Classes

- class BusOperatorFactory

  A singleton factory object, that can produce and return back the current IBusOperator object. This allows for support of several bus operators, not just Reading Buses. You would set the operator you want to get data for and the factory will then return a singleton reference to the operator object.
- class BusTimeTableStub

  Used when you need to problematically make a new IBusTimetable record, for example when you are making a new timetable during the evaluator.
- class CacheWriter

  Used to write a cache files to the disk.
- interface IBusHistoricTimeTable

  Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop.
- interface IBusOperator

  Provides all the information about a bus operator and gives the ability to query it further.
- interface IBusService

  A class which represents a single bus service.
- interface IBusSolidHistoricTimeTable

Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop. Solid, contains only records that had actual values recorded. Any non-solid record does not necessarily contain values.

- interface IBusStop

  A class which represents a single bus stop.
- interface IBusTimeTable

  A class which represents a single time table record, this is a time and particular bus stop for one service.
- class StubOperator

  The default operator if none is selected, this is mainly used to satisfy the null-ability requirement of C# then anything else.

Enumerations

- enum class Direction { **Inbound** , **Outbound** , **Both** }

  used to state the direction of travel of a service.
- enum class Operators { **ReadingBuses** }

  An enum which stores all the bus operators that the program supports.

4.2.1 Enumeration Type Documentation

4.2.1.1 Direction

`enum Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Direction [strong]`

used to state the direction of travel of a service.

4.2.1.2 Operators

`enum Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Operators [strong]`

An enum which stores all the bus operators that the program supports.

The program has been designed such that with only minimal changes any other bus operator API could be used instead with ease.

4.3 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.ReadingBuses Namespace Reference

Classes

- class **InternalCache**

  The Internal Cache class is used to store objects of type T, into program memory this is done for the best possible performance, while also ensuring that we don't exhaust all of a devices memory.
- class RbBusOperator

  An Implementation of the IBusOperator interface for the Reading Buses API.
- class RbBusService

  An Implementation of the IBusService interface for the Reading Buses API.
- class RbBusStop

  An Implementation of the IBusStop interface for the Reading Buses API.
- class RbTimeTable

  An Implementation of the IBusTimetable interface for the Reading Buses API.
- class RbTimeTableHistoric

  An Implementation of the IBusHistoricTimetable interface for the Reading Buses API.
- class RbTimeTableSolidHistoric

  An Implementation of the IBusSolidHistoricTimeTable interface for the Reading Buses API.

4.4 Timetable_Optimisation_Recommendations.Pages Namespace Reference

Classes

- class DateSelector

  The date selector page, this is where the user is asked to input a a date range, for which they intend upon making a new timetable for.
- class Evaluator

  The actual main evaluator stage of the program, this is the GUI for the "main part". But the page mainly contains the GUI to let the user know of the progress.
- class MainPage

  The main start page for the program, this is where the user will select the primary service that they wish to optimize for.
- class MovesMade

  Used to display the moves that have been made by the search algorithm after it has completed and displays the finalized timetables.
- class PreliminaryDataDownload

  The preliminary data download lets you download all of the data files that you are going to need for the search. It also lets you see the services past performance.
- class PreviousPerformance

  Once all of the data has been downloaded in the pre-evaluator checks show the performance metrics to the user. This is the final stage before starting the actual search.
- class RouteSegmentSelector

  The route-segment selector page finds other services that share a common route-segment with the primary service. And then allows the user to accept secondary services to also optimism their timetables for.

4.5 Timetable_Optimisation_Recommendations.Properties Namespace Reference

Classes

- class Resources

  A strongly-typed resource class, for looking up localized strings, etc.
- class **Settings**

4.6 Timetable_Optimisation_Recommendations.Request_Manager Namespace Reference

Classes

- class TimetableRetrieval

  Used to get a batch/ multiple-days worth of timetable data in a single query. However, this still has to be done sequentially because the Reading Buses API doesn't like having more than one database connection open at once.

4.7 Timetable_Optimisation_Recommendations.Route_Analyser Namespace Reference

Classes

- class RouteSegment

  Route Segment is set of consecutive stops that two services share, the primary service, identified in the Route↩ SegmenetFinder and the secondary service that also shares it.
- class RouteSegmentCollection

  The RouteSegmentCollection class manages the results of the RouteSegmentFinder, and provides the logic for part of the GUI, which lets the user add or remove a service of interest.
- class RouteSegmentFinder

  The route segment finder class takes in a Bus Service (known as the primary service) and a tolerance. It then finds any service which shares a common-route segment with it, to the specified minimum segment length tolerance.

4.8 Timetable_Optimisation_Recommendations.Search_Algorithm Namespace Reference

4.9 Timetable_Optimisation_Recommendations.Search_Algorithm.↩ Tabu_Search Namespace Reference

Classes

- struct Move

Used to represent a single move of the search algorithm, this involves making a change to one services timetable, on one running-board. A single timetable record is moved and then the surrounding records will also need to be edited in forwards and backwards propagation.

- class MoveSelector

  The move-selector is used to select the best move out of an array of moves, the neighborhood, to generate a new solution.
- class NeighbourhoodSolution

  Used to generate the neighbourhood of solutions for the search algorithm.
- class Solution

  Used to represent a single solution to the problem, the actual solution is stored in a dictionary, where the service is the key and an array of timetable records is the value.
- class TabuList

  TabuList keeps track of what moves are tabu/invalid and what moves can now be performed. Moves are tabu while the tabu tenure is greater than zero.

4.10 Timetable_Optimisation_Recommendations.Timetable_Analyser Namespace Reference

Classes

- class Cluster

  A cluster is a selection of dates which all have the same timetable. A group is a span of consecutive days within the cluster, that all have this same timetable.
- class Group

  A group is a collection of consecutive days within a cluster.
- class TimeTableGrouper

  The grouper class can be used to help find patterns in the timetable where several days shared the same timetable.

4.11 Timetable_Optimisation_Recommendations.Timetable_Evaluator Namespace Reference

Classes

- class BlamedBusTimeTable

  A "blamed" bus timetable is a timetable record that also contains weights and blame values, from squeaky wheel optimization. Along with the logic for updating the scheduled times.
- class PreEvaluatorChecks

  Pre-evaluator checks is run before actually running the real evaluator and is used to download all of the data that is required and evaluate the performance of the old timetable.
- class ServiceCohesionEvaluator

  The service cohesion evaluator works with the Evaluator class to calculate how well a services timetable works, with another service that shares a common route segment. By assigning a blame value to each timetable record based on how un-cohesive it is.
- class SlackTimeEvaluator

  The slack time evaluator works with the Evaluator class to calculate if the timetable has excessive slack (Or not enough). By assigning a blame value to each timetable record based on how disruptive it is to the timetable.
- class TimeTableEvaluator

  The main evaluator entry point, give it a set of dates where you want to use data from, an initial starting solution and a route-segment collection and then it can improve upon the timetable.
- class Weights

  The weights class is used to store a target arrival and departure time, along with a raw and standardised weight for how much it should pull towards it.

4.12 Timetable_Optimisation_Recommendations.Timetable_↩ Performance_Evaluator Namespace Reference

Classes

- class LatenessRecord

  Used to represent the lateness of one single record.
- struct LatenessReport

  Used for MVVM to bind to the GUI, this contains data summerising the performance of a service.
- class PerformanceEvaluator

  A class used to generate the performance of the historical/current timetables.

4.13 Timetable_Optimisation_Recommendations.Timetable_Simulator Namespace Reference

Classes

- class DwellTimeSimulator

  Used to calculate how long a bus is going to need to dwell at a bus stop, given the time of day and hence changes in passenger demand. Can be considered how busy a stop is.
- class JourneyTime

  A simplistic class used to help represent the journey time between two stops.
- class JourneyTimeSimulator

  The journey time simulator class takes in a theoretical departure time, two stops to travel between and some other information. It then estimates how long it would likely take to travel between the two stops at this time of day.
- class TimeSimulator

  Shared common code used between both the Journey Time Simulator and the Dwell Time Simulator. Manila the weighted averaging code and accuracy measures.

4.14 Timetable_Optimisation_Recommendations.UserControls Namespace Reference

Classes

- class ClusterCard

  The card used to show a timetable cluster and the groupings within it.
- class LatenessSummary

  The lateness summary report card, shows the performance of the service historically.
- class NotifyTaskCompletion

  Helper class to wrap around a Task to provide more information usable for UI databinding scenarios. As discussed in MSDN Magazine: https://msdn.microsoft.com/magazine/dn605875.
- class ServiceCard

  The service card, use to display the single service.

4.15 Timetable_Optimisation_Recommendations.Windows Namespace Reference

Classes

- class Highlight

  Used to tell the timetable data grid what colour highlights the cells should be.
- class MainWindow

  The main entry point for the application, but the logic is contained within the page.
- class Settings

  The main settings page for the application.
- class ViewTimetable

  Used to display the timetable to the user.
- class ViewTimetableHighlighted

  Interaction logic for ViewTimetable.xaml

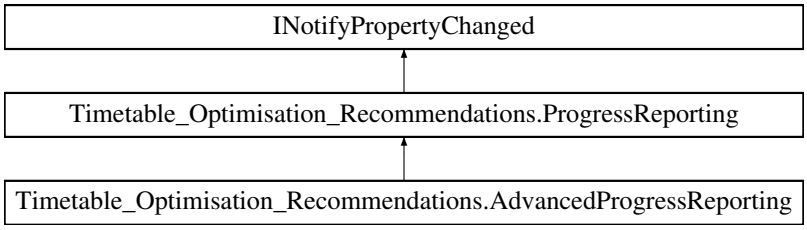
Chapter 5

Class Documentation

5.1 Timetable_Optimisation_Recommendations.AdvancedProgressReporting Class Reference

Advanced Progress Reporting, is used to report back to the GUI on the progress of a more complex task that contains sub-tasks.

Inheritance diagram for Timetable_Optimisation_Recommendations.AdvancedProgressReporting:



Public Member Functions

- AdvancedProgressReporting (double value, double subValue, string message)

  The default constructor for the advanced reporter.
- AdvancedProgressReporting (string stage, int stageVal)

  A constructor used to update the current stage message and value.
- AdvancedProgressReporting ()

  The default constructor for the class.
- void Update (AdvancedProgressReporting reporter)

  Used to update an advanced progress reporter with another object.
- new void Clear ()

  Used to reset the progress back down to nothing/zero.

Properties

- double SubValue `[get, set]`

  The progress of the sub value of the task.
- string? Stage `[get, set]`

  value>The Stage integer value it is on. Total number of stages isn't specified but normally three.
- int **StageVal** `[get, set]`

Additional Inherited Members

5.1.1 Detailed Description

Advanced Progress Reporting, is used to report back to the GUI on the progress of a more complex task that contains sub-tasks.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 AdvancedProgressReporting() [1/3]

```
Timetable_Optimisation_Recommendations.AdvancedProgressReporting.AdvancedProgressReporting (
            double value,
            double subValue,
            string message )
```

The default constructor for the advanced reporter.

Parameters

value	The value of the overall task.
subValue	The value of the sub-task.
message	A message to say what was last completed.

5.1.2.2 AdvancedProgressReporting() [2/3]

```
Timetable_Optimisation_Recommendations.AdvancedProgressReporting.AdvancedProgressReporting (
            string stage,
            int stageVal )
```

A constructor used to update the current stage message and value.

Parameters

stage	
stageVal	

5.1.2.3 AdvancedProgressReporting() [3/3]

```
Timetable_Optimisation_Recommendations.AdvancedProgressReporting.AdvancedProgressReporting ( )
```

The default constructor for the class.

5.1.3 Member Function Documentation

5.1.3.1 Clear()

```
new void Timetable_Optimisation_Recommendations.AdvancedProgressReporting.Clear ( )
```

Used to reset the progress back down to nothing/zero.

5.1.3.2 Update()

```
void Timetable_Optimisation_Recommendations.AdvancedProgressReporting.Update (
            AdvancedProgressReporting reporter )
```

Used to update an advanced progress reporter with another object.

Parameters

reporter	

5.1.4 Property Documentation

5.1.4.1 SubValue

```
double Timetable_Optimisation_Recommendations.AdvancedProgressReporting.SubValue  [get], [set]
```

The progress of the sub value of the task.

value>The Stage name of the task is currently on (if any)

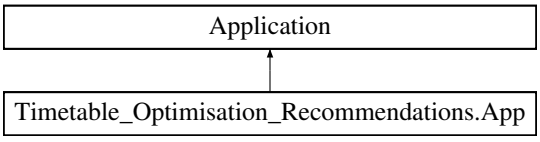
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/AdvancedProgress↩ Reporting.cs

5.2 Timetable_Optimisation_Recommendations.App Class Reference

Interaction logic for App.xaml

Inheritance diagram for Timetable_Optimisation_Recommendations.App:



Protected Member Functions

- override async void **OnStartup** (StartupEventArgs e)

5.2.1 Detailed Description

Interaction logic for App.xaml

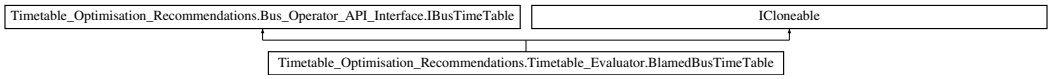
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/App.xaml.cs

5.3 Timetable_Optimisation_Recommendations.Timetable_Evaluator.↩ BlamedBusTimeTable Class Reference

A "blamed" bus timetable is a timetable record that also contains weights and blame values, from squeaky wheel optimization. Along with the logic for updating the scheduled times.

Inheritance diagram for Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable:



Public Member Functions

- BlamedBusTimeTable (IBusTimeTable record)

  The default constructor for the blamed record timetable, takes in a pre-existing timetable record.
- void SetSuggestedToReal ()

  Updates the Scheduled Arrival and Departure times to their suggested values calculated from their blame values in SWO. This should only be called on the target record.
- DateTime ProposedSchArrivalTime ()

  Calculates the proposed scheduled arrival time. Uses the blame values and weights to suggest a new time.
- DateTime ProposedSchDepartureTime ()

  Calculates the proposed scheduled departure time. Uses the blame values and weights to suggest a new time.
- object Clone ()

  Performs a deep clone of the object.
- void UpdateTimes (DateTime arrival, DateTime departure)

  Given a new arrival and departure date update it accordingly within the record.
- void UpdateTotalWeight ()

  Generates the total weight, we do not need to apply dominance alterations here as we have already applied the dominance earlier on.
- bool MatchDirection (Direction direction)

  Used to check if the direction of travel of this record matches the value or not.
- bool WeakIsStopSame (IBusTimeTable stop2)

  A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.
- bool WeakIsStopSame (IBusStop stop2)

  Used to say that if given a bus stop object is the stop about this timetable record or not.
- override bool **Equals** (object obj)
- bool **Equals** (BlamedBusTimeTable? other)
- override int **GetHashCode** ()
- string GetId ()

  Used to get a unique ID value to represent the timetable record.

Properties

- Weights SlackWeights = new () `[get]`

  Stores the blame values for the slack time objective.
- Weights CohesionWeights = new () `[get]`

  value>Stores the total blame for the record as a whole.
- double TotalWeight = 0 `[get]`

  value>Stores the Scheduled Arrival Time
- DateTime SchArrivalTime `[get]`

  value>Stores the Scheduled Departure Time
- DateTime SchDepartureTime `[get]`

  value>A reference is kept to the original record as lazy evaluation means it would be impractical to deep clone.
- IBusStop Location `[get]`

  The rest of the below is your standard code for a bus timetable record.
- long **Sequence** `[get]`
- bool **IsOutbound** `[get]`
- string **JourneyCode** `[get]`
- string **RunningBoard** `[get]`
- bool **IsTimingPoint** `[get]`
- IBusService **Service** `[get]`

5.3.1 Detailed Description

A "blamed" bus timetable is a timetable record that also contains weights and blame values, from squeaky wheel optimization. Along with the logic for updating the scheduled times.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 BlamedBusTimeTable()

```
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.BlamedBusTime↩
Table (
            IBusTimeTable record )
```

The default constructor for the blamed record timetable, takes in a pre-existing timetable record.

Parameters

record	The pre-existing timetable record.

5.3.3 Member Function Documentation

5.3.3.1 Clone()

```
object Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.Clone ( )
```

Performs a deep clone of the object.

Returns

A deep clone of the current object,

5.3.3.2 GetId()

```
string Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.GetId ( )
```

Used to get a unique ID value to represent the timetable record.

Returns

A value to represent the record.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.3.3.3 MatchDirection()

```
bool Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.Match↩
Direction (
            Direction direction )
```

Used to check if the direction of travel of this record matches the value or not.

Parameters

direction	

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.3.3.4 ProposedSchArrivalTime()

```
DateTime Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.↩
ProposedSchArrivalTime ( )
```

Calculates the proposed scheduled arrival time. Uses the blame values and weights to suggest a new time.

Returns

The best new arrival time for this record

5.3.3.5 ProposedSchDepartureTime()

```
DateTime Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.↩
ProposedSchDepartureTime ( )
```

Calculates the proposed scheduled departure time. Uses the blame values and weights to suggest a new time.

Returns

The best new departure time for this record.

5.3.3.6 SetSuggestedToReal()

```
void Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.SetSuggested↩
ToReal ( )
```

Updates the Scheduled Arrival and Departure times to their suggested values calculated from their blame values in SWO. This should only be called on the target record.

5.3.3.7 UpdateTimes()

```
void Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.UpdateTimes
(
            DateTime arrival,
            DateTime departure )
```

Given a new arrival and departure date update it accordingly within the record.

Parameters

arrival	new arrival time.
departure	new departure time.

5.3.3.8 UpdateTotalWeight()

```
void Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.Update↩
TotalWeight ( )
```

Generates the total weight, we do not need to apply dominance alterations here as we have already applied the dominance earlier on.

This is the normalized weights added together.

5.3.3.9 WeakIsStopSame() [1/2]

```
bool Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.WeakIs↩
StopSame (
            IBusStop stop2 )
```

Used to say that if given a bus stop object is the stop about this timetable record or not.

Parameters

stop2	

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.3.3.10 WeakIsStopSame() [2/2]

```
bool Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.WeakIs↩
StopSame (
            IBusTimeTable stop2 )
```

A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.

Parameters

stop2	Another time table record to compare against.

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.3.4 Property Documentation

5.3.4.1 Location

```
IBusStop Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.↩
Location  [get]
```

The rest of the below is your standard code for a bus timetable record.

5.3.4.2 SlackWeights

```
Weights Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.Slack↩
Weights = new ()  [get]
```

Stores the blame values for the slack time objective.

value>Stores the blame values for the cohesion objective.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Evaluator/Blamed↩ BusTimeTable.cs

5.4 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.BusOperatorFactory Class Reference

A singleton factory object, that can produce and return back the current IBusOperator object. This allows for support of several bus operators, not just Reading Buses. You would set the operator you want to get data for and the factory will then return a singleton reference to the operator object.

Public Member Functions

- async Task SetOperatorAsync (Operators selected)

  Used to switch between operators to return.

Properties

- static BusOperatorFactory Instance `[get]`

  Used to request an instance to the singleton object.
- IBusOperator Operator = new StubOperator() `[get]`

  Stores the object reference for the current operator.

5.4.1 Detailed Description

A singleton factory object, that can produce and return back the current IBusOperator object. This allows for support of several bus operators, not just Reading Buses. You would set the operator you want to get data for and the factory will then return a singleton reference to the operator object.

5.4.2 Member Function Documentation

5.4.2.1 SetOperatorAsync()

```
async Task Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperator↩
Factory.SetOperatorAsync (
            Operators selected )
```

Used to switch between operators to return.

Parameters

selected	The bus operator to return

Returns

Nothing, signifies once completed.

5.4.3 Property Documentation

5.4.3.1 Instance

```
BusOperatorFactory Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Bus↩
OperatorFactory.Instance  [static], [get]
```

Used to request an instance to the singleton object.

5.4.3.2 Operator

```
IBusOperator Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperator↩
Factory.Operator = new StubOperator()  [get]
```

Stores the object reference for the current operator.

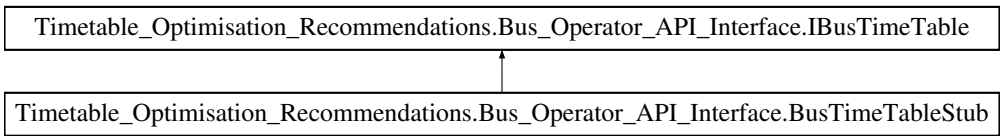
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/BusOperatorFactory.cs

5.5 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.BusTimeTableStub Class Reference

Used when you need to problematically make a new IBusTimetable record, for example when you are making a new timetable during the evaluator.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTable↩ Stub:



Public Member Functions

- **BusTimeTableStub** (IBusTimeTable timeTable)
- bool MatchDirection (Direction direction)

  Used to check if the direction of travel of this record matches the value or not.
- **BusTimeTableStub** (IBusTimeTable timeTable, TimeSpan schArrivalTime, TimeSpan schDepartureTime)
- bool WeakIsStopSame (IBusTimeTable stop2)

  A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.
- bool WeakIsStopSame (IBusStop stop2)

  Used to say that if given a bus stop object is the stop about this timetable record or not.
- string GetId ()

  Used to get a unique ID value to represent the timetable record.

Properties

- long **Sequence** `[get, protected set]`
- bool **IsOutbound** `[get, protected set]`
- string **JourneyCode** `[get, protected set]`
- bool **IsTimingPoint** `[get, protected set]`
- DateTime **SchArrivalTime** `[get, set]`
- DateTime **SchDepartureTime** `[get, set]`
- string **ServiceId** `[get, set]`
- string **StopId** `[get, set]`
- string **RunningBoard** `[get, set]`
- IBusStop **Location** `[get]`
- IBusService **Service** `[get]`

5.5.1 Detailed Description

Used when you need to problematically make a new IBusTimetable record, for example when you are making a new timetable during the evaluator.

5.5.2 Member Function Documentation

5.5.2.1 GetId()

```
string Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.↩
GetId ( )
```

Used to get a unique ID value to represent the timetable record.

Returns

A value to represent the record.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.5.2.2 MatchDirection()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.↩
MatchDirection (
            Direction direction )
```

Used to check if the direction of travel of this record matches the value or not.

Parameters

direction	

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.5.2.3 WeakIsStopSame() [1/2]

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.Weak↩
IsStopSame (
            IBusStop stop2 )
```

Used to say that if given a bus stop object is the stop about this timetable record or not.

Parameters

stop2	

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.5.2.4 WeakIsStopSame() [2/2]

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.Weak↩
IsStopSame (
            IBusTimeTable stop2 )
```

A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.

Parameters

stop2	Another time table record to compare against.

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/BusTimeTableStub.cs

5.6 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.CacheWriter Class Reference

Used to write a cache files to the disk.

Static Public Member Functions

- static void WriteToCache (string fileLoc, string fileName, string? content)

  Saves a cache file to local disk, and hides the folder so the user cannot see it.

5.6.1 Detailed Description

Used to write a cache files to the disk.

5.6.2 Member Function Documentation

5.6.2.1 WriteToCache()

```
static void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.CacheWriter.↩
WriteToCache (
            string fileLoc,
            string fileName,
            string?  content )  [static]
```

Saves a cache file to local disk, and hides the folder so the user cannot see it.

Parameters

fileLoc	The location for the cache file.
fileName	The name of the file
content	The contents of the file.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/CacheWriter.cs

5.7 Timetable_Optimisation_Recommendations.Timetable_Analyser.↩ Cluster Class Reference

A cluster is a selection of dates which all have the same timetable. A group is a span of consecutive days within the cluster, that all have this same timetable.

Public Member Functions

- Cluster (DateTime date, IBusTimeTable[ ] timeTable)

  The default constructor for the cluster. Given one date for each the timetable applies and then the timetable associated.
- void AddDate (DateTime date)

  Used to associate another date with the cluster.
- void CalculateGrouping ()

  Builds up the groups from the cluster.
- IBusService GetAssociatedService ()

  Gets the associated service for the cluster.

Properties

- List< DateTime > AssociatedTimes [get]

  A list of dates which are associated with this timetable, i.e they had the same timetable as each other.
- IBusTimeTable[ ] BusTimeTables [get]

  The timetable for this cluster, an array of records for one day.
- Group? GroupingAssociated [get]

  The Grouping associated with the cluster.
- int ClusterId [get]

  A unique ID for the cluster.

5.7.1 Detailed Description

A cluster is a selection of dates which all have the same timetable. A group is a span of consecutive days within the cluster, that all have this same timetable.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 Cluster()

```
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.Cluster (
            DateTime date,
            IBusTimeTable[] timeTable )
```

The default constructor for the cluster. Given one date for each the timetable applies and then the timetable associated.

Parameters

date	A date associated with the timetable.
timeTable	The timetable it self.

5.7.3 Member Function Documentation

5.7.3.1 AddDate()

```
void Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.AddDate (
            DateTime date )
```

Used to associate another date with the cluster.

Parameters

date	A new date to add to the cluster.

5.7.3.2 CalculateGrouping()

```
void Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.CalculateGrouping ( )
```

Builds up the groups from the cluster.

5.7.3.3 GetAssociatedService()

```
IBusService Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.GetAssociated↩
Service ( )
```

Gets the associated service for the cluster.

Returns

5.7.4 Property Documentation

5.7.4.1 AssociatedTimes

```
List<DateTime> Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.Associated↩
Times  [get]
```

A list of dates which are associated with this timetable, i.e they had the same timetable as each other.

5.7.4.2 BusTimeTables

```
IBusTimeTable [] Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.BusTime↩
Tables  [get]
```

The timetable for this cluster, an array of records for one day.

5.7.4.3 ClusterId

```
int Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.ClusterId  [get]
```

A unique ID for the cluster.

5.7.4.4 GroupingAssociated

```
Group?  Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster.GroupingAssociated
[get]
```

The Grouping associated with the cluster.

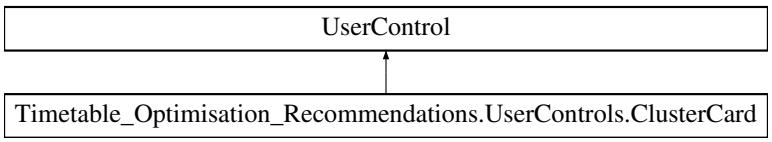
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Analyser/Cluster.cs

5.8 Timetable_Optimisation_Recommendations.UserControls.ClusterCard Class Reference

The card used to show a timetable cluster and the groupings within it.

Inheritance diagram for Timetable_Optimisation_Recommendations.UserControls.ClusterCard:



Public Member Functions

- ClusterCard ()
  The default constructor.

5.8.1 Detailed Description

The card used to show a timetable cluster and the groupings within it.

5.8.2 Constructor & Destructor Documentation

5.8.2.1 ClusterCard()

```
Timetable_Optimisation_Recommendations.UserControls.ClusterCard.ClusterCard ( )
```

The default constructor.

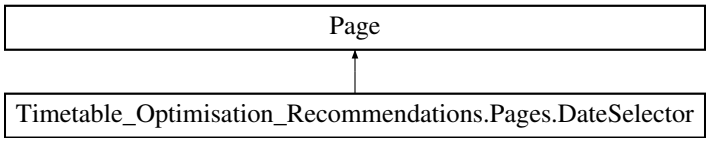
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/UserControls/ClusterCard.xaml.cs

5.9 Timetable_Optimisation_Recommendations.Pages.DateSelector Class Reference

The date selector page, this is where the user is asked to input a a date range, for which they intend upon making a new timetable for.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.DateSelector:



Public Member Functions

- DateSelector (string serviceId)

  The default constructor, takes in a service ID string and then creates the date selector for that service.

Properties

- DateTime StartDate = DateTime.Today `[get, set]`

  The Start date for which they wish to get timetable data for.
- DateTime EndDate = DateTime.Today `[get, set]`

  value>The service for which they are finding timetables groups for.
- IBusService Service `[get]`

  value>Used to store when different timetables were in affect.
- ObservableCollection< Cluster > TimeTableClusters = new() `[get]`

  value>Used to update the GUI progress bar.
- ProgressReporting **Reporter** = new() `[get]`

5.9.1 Detailed Description

The date selector page, this is where the user is asked to input a a date range, for which they intend upon making a new timetable for.

5.9.2 Constructor & Destructor Documentation

5.9.2.1 DateSelector()

```
Timetable_Optimisation_Recommendations.Pages.DateSelector.DateSelector (
            string serviceId )
```

The default constructor, takes in a service ID string and then creates the date selector for that service.

Parameters

serviceId	

5.9.3 Property Documentation

5.9.3.1 StartDate

```
DateTime Timetable_Optimisation_Recommendations.Pages.DateSelector.StartDate = DateTime.Today
[get], [set]
```

The Start date for which they wish to get timetable data for.

value>The end date for which they wish to get timetable data for.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/DateSelector.xaml.cs

5.10 Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan Class Reference

A class used to represent a date span, which is a period between two dates.

Public Member Functions

- DateSpan (DateTime start, DateTime end)

  A default constructor, which takes in an initial start and end date for the date span.
- DateSpan (DateTime start)

  As the start and end initial values are likely to be the same. Until new data can be found, this constructor sets both to be the same.
- int TotalSpan ()

  How long the span is, 0 for one day.
- void AddDate (DateTime date)

  Adds a new date to the date span, this date must be one newer or one day older than the current span. A span must be consecutive, so you cannot adjust the start or end span several days off the previous known cluster.
- bool IsWeekDay ()

  Returns if the date-span is only contained within a weekday (Monday to Friday) Or if it contains weekends as well. This is because I'm assuming weekdays will have generally contestant timetables.
- override string ToString ()

  The string representation of the group.

Properties

- DateTime Start `[get]`

  The start date of the span, this should be the oldest of the dates.
- DateTime End `[get]`

  The end date of the span, this should the newest of the dates.

5.10.1 Detailed Description

A class used to represent a date span, which is a period between two dates.

5.10.2 Constructor & Destructor Documentation

5.10.2.1 DateSpan() [1/2]

```
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.DateSpan (
            DateTime start,
            DateTime end )
```

A default constructor, which takes in an initial start and end date for the date span.

Parameters

start	An starting value for the date span.
end	An ending value for the date span.

5.10.2.2 DateSpan() [2/2]

```
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.DateSpan (
            DateTime start )
```

As the start and end initial values are likely to be the same. Until new data can be found, this constructor sets both to be the same.

Parameters

start	The start and end date for the date span

5.10.3 Member Function Documentation

5.10.3.1 AddDate()

```
void Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.AddDate (
            DateTime date )
```

Adds a new date to the date span, this date must be one newer or one day older than the current span. A span must be consecutive, so you cannot adjust the start or end span several days off the previous known cluster.

Parameters

date	A new start or end date for the span.

5.10.3.2 IsWeekDay()

```
bool Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.IsWeekDay ( )
```

Returns if the date-span is only contained within a weekday (Monday to Friday) Or if it contains weekends as well. This is because I'm assuming weekdays will have generally contestant timetables.

Returns

5.10.3.3 ToString()

```
override string Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.To↩
String ( )
```

The string representation of the group.

Returns

5.10.3.4 TotalSpan()

```
int Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.TotalSpan ( )
```

How long the span is, 0 for one day.

Returns

How long the date span is.

5.10.4 Property Documentation

5.10.4.1 End

```
DateTime Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.End  [get]
```

The end date of the span, this should the newest of the dates.

5.10.4.2 Start

```
DateTime Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan.Start  [get]
```

The start date of the span, this should be the oldest of the dates.

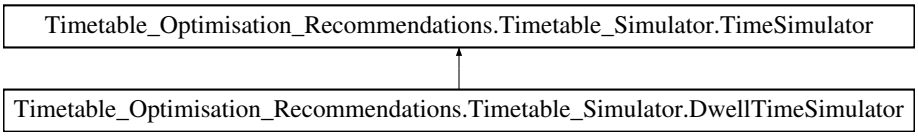
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Analyser/Group.↩
cs

5.11 Timetable_Optimisation_Recommendations.Timetable_Simulator.↩ DwellTimeSimulator Class Reference

Used to calculate how long a bus is going to need to dwell at a bus stop, given the time of day and hence changes in passenger demand. Can be considered how busy a stop is.

Inheritance diagram for Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeSimulator:



Public Member Functions

- DwellTimeSimulator (DateTime[ ] cluster, IBusStop busStop, TimeSpan time, IBusService[ ] services)

  The default constructor for the class.
- async Task< TimeSpan > ProduceEstimatedDwell (IProgress< double >? progress)

  Actually generates the time that is estimated for the bus to dwell at the stop.

Additional Inherited Members

5.11.1 Detailed Description

Used to calculate how long a bus is going to need to dwell at a bus stop, given the time of day and hence changes in passenger demand. Can be considered how busy a stop is.

5.11.2 Constructor & Destructor Documentation

5.11.2.1 DwellTimeSimulator()

```
Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeSimulator.DwellTime↩
Simulator (
            DateTime[] cluster,
            IBusStop busStop,
            TimeSpan time,
            IBusService[] services )
```

The default constructor for the class.

Parameters

cluster	Dates for which we can request data from
busStop	The bus stop in question to get dwell time for
time	The time a bus is meant to arrive at the stop
services	What bus services stopping at the stop were interested in

5.11.3 Member Function Documentation

5.11.3.1 ProduceEstimatedDwell()

```
async Task<TimeSpan> Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTime↩
Simulator.ProduceEstimatedDwell (
            IProgress< double >?  progress )
```

Actually generates the time that is estimated for the bus to dwell at the stop.

Parameters

progress	Progress bar used to update the GUI on the progress of the task.

Returns

Given the starting parameters the final outputted dwell time estimate.

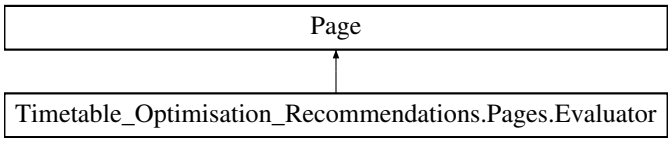
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Simulator/Dwell↩ TimeSimulator.cs

5.12 Timetable_Optimisation_Recommendations.Pages.Evaluator Class Reference

The actual main evaluator stage of the program, this is the GUI for the "main part". But the page mainly contains the GUI to let the user know of the progress.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.Evaluator:



Public Member Functions

- Evaluator (PreEvaluatorChecks preEvaluator)

  The default evaluator constructor, takes in the information from the pre-evaluator.

Properties

- AdvancedProgressReporting Reporter = new() `[get]`

  Used to report back to GUI the process progress.

5.12.1 Detailed Description

The actual main evaluator stage of the program, this is the GUI for the "main part". But the page mainly contains the GUI to let the user know of the progress.

5.12.2 Constructor & Destructor Documentation

5.12.2.1 Evaluator()

```
Timetable_Optimisation_Recommendations.Pages.Evaluator.Evaluator (
            PreEvaluatorChecks preEvaluator )
```

The default evaluator constructor, takes in the information from the pre-evaluator.

Parameters

preEvaluator	The pre-evaluator object

5.12.3 Property Documentation

5.12.3.1 Reporter

```
AdvancedProgressReporting Timetable_Optimisation_Recommendations.Pages.Evaluator.Reporter =
new()  [get]
```

Used to report back to GUI the process progress.

value>Contains the actual logic and implementation of the evaluator.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/Evaluator.↩ xaml.cs

5.13 Timetable_Optimisation_Recommendations.Timetable_Analyser.↩ Group Class Reference

A group is a collection of consecutive days within a cluster.

Classes

- class DateSpan

  A class used to represent a date span, which is a period between two dates.

Public Member Functions

- Group (List< DateTime > associatedTimes)

  The default constructor for the group.
- List< DateSpan > GroupingsOfClusters (List< DateTime > associatedTimes)

  Generates a grouping within the cluster. A group is a set of consecutive days which share the same timetable. This finds all the groups within the cluster and adds it to a list. A group can be one single day.
- async Task< string > GetStringAsync (List< DateTime > associatedTimes)

  Generates the string for the GUI to output.

Properties

- List< DateSpan >? Grouping `[get]`

  A list of date spans, which contains the groups where times were the same.
- NotifyTaskCompletion< string > List `[get]`

  value>Contains the GUI grouping representation.

5.13.1 Detailed Description

A group is a collection of consecutive days within a cluster.

5.13.2 Constructor & Destructor Documentation

5.13.2.1 Group()

```
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.Group (
            List< DateTime > associatedTimes )
```

The default constructor for the group.

Parameters

associatedTimes	Takes in just a list of date times to group.

5.13.3 Member Function Documentation

5.13.3.1 GetStringAsync()

```
async Task<string> Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.GetString↩
Async (
            List< DateTime > associatedTimes )
```

Generates the string for the GUI to output.

Parameters

associatedTimes	A list of date times to group.

Returns

A string summarizing the groups that have been found.

5.13.3.2 GroupingsOfClusters()

```
List<DateSpan> Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.GroupingsOf↩
Clusters (
            List< DateTime > associatedTimes )
```

Generates a grouping within the cluster. A group is a set of consecutive days which share the same timetable. This finds all the groups within the cluster and adds it to a list. A group can be one single day.

Returns

Groupings within a cluster.

5.13.4 Property Documentation

5.13.4.1 Grouping

```
List<DateSpan>?  Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.Grouping
[get]
```

A list of date spans, which contains the groups where times were the same.

value>Used by the GUI to know the groups and how/when to update.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Analyser/Group.↩ cs

5.14 Timetable_Optimisation_Recommendations.Windows.Highlight Class Reference

Used to tell the timetable data grid what colour highlights the cells should be.

Properties

- int X `[get]`

  The X coordinate in the grid.
- int Y `[get]`

  value>Is it for the outbound or inbound table.
- bool IsOutbound `[get]`

  value>The total weighting cell colour.
- SolidColorBrush TotalWeight = Brushes.Green `[get]`

  value>The slack time cell colour
- SolidColorBrush SlackWeight = Brushes.Green `[get]`

  value>The cohesion value cell colour
- SolidColorBrush CohesionWeight = Brushes.Green `[get]`

  value>If it's a moved record or not cell colour.
- SolidColorBrush **MoveHighlight** = new SolidColorBrush(Color.FromArgb(0, 255, 255, 255)) `[get, set]`

5.14.1 Detailed Description

Used to tell the timetable data grid what colour highlights the cells should be.

5.14.2 Property Documentation

5.14.2.1 X

```
int Timetable_Optimisation_Recommendations.Windows.Highlight.X [get]
```

The X coordinate in the grid.

value>The Y coordinate in the grid.

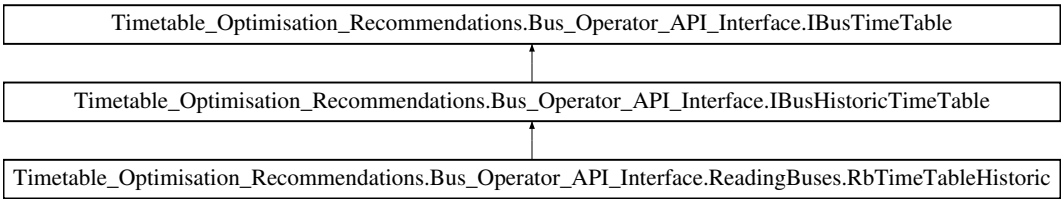
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Windows/Highlight.cs

5.15 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.IBusHistoricTimeTable Interface Reference

Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoric↩ TimeTable:



Public Member Functions

- bool CouldBeSolid ()

  Returns if the record could be made a "solid" record or not. A solid record is one with reported arrival and departure times.
- IBusSolidHistoricTimeTable GetSolid ()

  Gets the solid representation of the same object.

Properties

- DateTime? ActArrivalTime `[get]`

  The actual arrival time for the bus.
- DateTime? ActDepartureTime `[get]`

  The actual departure time for the bus.

5.15.1 Detailed Description

Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop.

5.15.2 Member Function Documentation

5.15.2.1 CouldBeSolid()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoricTime↩
Table.CouldBeSolid ( )
```

Returns if the record could be made a "solid" record or not. A solid record is one with reported arrival and departure times.

Returns

true if Actual Arrival and Departure have values.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableHistoric.

5.15.2.2 GetSolid()

```
IBusSolidHistoricTimeTable Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.↩
IBusHistoricTimeTable.GetSolid ( )
```

Gets the solid representation of the same object.

Returns

Gets the solid equivalence object.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableHistoric.

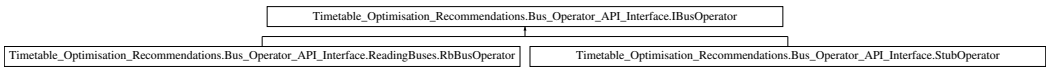
The documentation for this interface was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/IBusHistoricTimeTable.cs

5.16 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.IBusOperator Interface Reference

Provides all the information about a bus operator and gives the ability to query it further.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator:



Public Member Functions

- IBusService GetService (string serviceNumber)

  Returns a service which matches the Service Number passed.
- bool IsService (string serviceNumber)

  Checks to see if a service of that number exists or not in the API feed.
- IBusService[ ] GetServices ()

  Gets an array of all the IBusServices Objects.
- IBusStop GetLocation (string atcoCode)

  Get a bus stop location based upon a bus stops location code
- bool IsLocation (string atcoCode)

  Checks to see if the atco code for the bus stop exists in the API feed or not.
- void InvalidateCache ()

  Deletes any Cache data stored, use this only if you need to force new data cache.
- void ForceUpdateCache ()

  Forces the current data stored in the bus operator object to be saved into Cache. You would need to do this if you've made some lazy API requests.

5.16.1 Detailed Description

Provides all the information about a bus operator and gives the ability to query it further.

5.16.2 Member Function Documentation

5.16.2.1 ForceUpdateCache()

```
void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.Force↩
UpdateCache ( )
```

Forces the current data stored in the bus operator object to be saved into Cache. You would need to do this if you've made some lazy API requests.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator.

5.16.2.2 GetLocation()

```
IBusStop Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.Get↩
Location (
            string atcoCode )
```

Get a bus stop location based upon a bus stops location code

Parameters

atcoCode	The code of the bus stop

Returns

A Bus Stop object for the Atco Code specified.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator.

5.16.2.3 GetService()

```
IBusService Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.↩
GetService (
            string serviceNumber )
```

Returns a service which matches the Service Number passed.

Parameters

serviceNumber	The service number/ID for the service you wish to be returned eg: 17 or 22.

Returns

The services matching the ID.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator.

5.16.2.4 GetServices()

```
IBusService [] Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.↩
GetServices ( )
```

Gets an array of all the IBusServices Objects.

Returns

An array of all the bus services.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator.

5.16.2.5 InvalidateCache()

```
void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.Invalidate↩
Cache ( )
```

Deletes any Cache data stored, use this only if you need to force new data cache.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator.

5.16.2.6 IsLocation()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.IsLocation
(
            string atcoCode )
```

Checks to see if the atco code for the bus stop exists in the API feed or not.

Parameters

atcoCode	The ID Code for a bus stop.

Returns

True or False depending on if the stop is in the API feed or not.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.

5.16.2.7 IsService()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.IsService
(
            string serviceNumber )
```

Checks to see if a service of that number exists or not in the API feed.

Parameters

serviceNumber	The service number to find.

Returns

True or False for if a service is the API feed or not.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator.

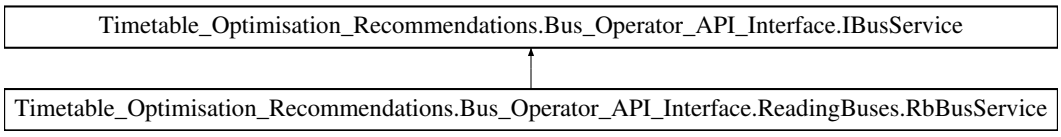
The documentation for this interface was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/IBusOperator.cs

5.17 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService Interface Reference

A class which represents a single bus service.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService:



Public Member Functions

- Task< IBusStop[ ]> GetLocations (Direction direction=Direction.Both)

  Gets an array of 'BusStop' objects the bus service travels too as an array of BusStop objects. If the API is invalid and links to a Bus Stop not in the list of locations it will simply be ignored.
- Task< IBusTimeTable[ ]?> GetTimeTable (DateTime date)

  Gets the planned timetable departure and arrival times for this service on a specific date.
- bool IsTimeTableCached (DateTime date)

  Tells you if a file has been cached or not on disk.
- Task< IBusHistoricTimeTable[ ]?> GetArchivedTimeTable (DateTime date)

  Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date.
- bool IsArchivedTimeTableCached (DateTime date)

  Tells you if a file has been cached or not on disk.
- bool IsWeakServiceSame (IBusService service)

  Given another IBusService Object, check if it is the same service or not. Only by comparing the service ID value.

Properties

- string ServiceId `[get]`

5.17.1 Detailed Description

A class which represents a single bus service.

5.17.2 Member Function Documentation

5.17.2.1 GetArchivedTimeTable()

```
Task<IBusHistoricTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_API_↩
Interface.IBusService.GetArchivedTimeTable (
            DateTime date )
```

Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date.

Parameters

date	the date on which you want a archived timetable data for. This should be a date in the past.

Returns

An array of time table records, containing the scheduled and actual arrival and departure times of buses.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService.

5.17.2.2 GetLocations()

```
Task<IBusStop[]> Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBus↩
Service.GetLocations (
            Direction direction = Direction.Both )
```

Gets an array of 'BusStop' objects the bus service travels too as an array of BusStop objects. If the API is invalid and links to a Bus Stop not in the list of locations it will simply be ignored.

Parameters

direction	Used to filter by the direction of travel the stops are on.

Returns

An array of BusStop objects for the stops visited by this service.

It is assumed that the ordering of the array is the ordering in which a service will visit all of the stops. If the ordering is incorrect the route-segment finder will fail.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService.

5.17.2.3 GetTimeTable()

```
Task<IBusTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.↩
IBusService.GetTimeTable (
            DateTime date )
```

Gets the planned timetable departure and arrival times for this service on a specific date.

Parameters

date	the date on which you want a archived timetable data for. This should be a date in the past.

Returns

An array of time table records, containing the planned scheduled and actual arrival and departure times of buses.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService.

5.17.2.4 IsArchivedTimeTableCached()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.IsArchived↩
TimeTableCached (
            DateTime date )
```

Tells you if a file has been cached or not on disk.

Parameters

date	The date for the time table date to search for.

Returns

True if the data is cached on disk

GetArchivedTimeTable(DateTime)

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService.

5.17.2.5 IsTimeTableCached()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.IsTime↩
TableCached (
            DateTime date )
```

Tells you if a file has been cached or not on disk.

Parameters

date	The date for the time table date to search for.

Returns

True if the data is cached on disk

GetTimeTable(DateTime)

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService.

5.17.2.6 IsWeakServiceSame()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.IsWeak↩
ServiceSame (
            IBusService service )
```

Given another IBusService Object, check if it is the same service or not. Only by comparing the service ID value.

Parameters

service	The other service you wish to compare against.

Returns

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService.

5.17.3 Property Documentation

5.17.3.1 ServiceId

```
string Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.ServiceId
[get]
```

The unique alphanumeric identifier for a bus service.

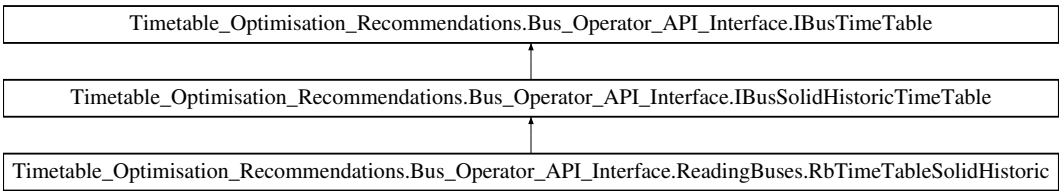
The documentation for this interface was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/IBusService.cs

5.18 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusSolidHistoricTimeTable Interface Reference

Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop. Solid, contains only records that had actual values recorded. Any non-solid record does not necessarily contain values.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusSolidHistoricTimeTable:



Properties

- DateTime ActArrivalTime `[get]`

  The actual arrival time for the bus.
- DateTime ActDepartureTime `[get]`

  The actual departure time for the bus.

Additional Inherited Members

5.18.1 Detailed Description

Used to store a historical time table record, which is an actual record for when a single bus arrive at a stop. Solid, contains only records that had actual values recorded. Any non-solid record does not necessarily contain values.

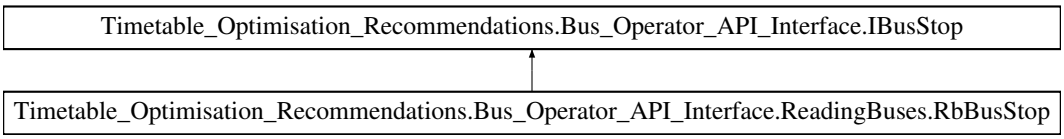
The documentation for this interface was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/IBusSolidHistoricTimeTable.cs

5.19 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.IBusStop Interface Reference

A class which represents a single bus stop.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop:



Public Member Functions

- IBusService[ ] GetServices ()

  Finds the 'BusService' object for all of the bus services which visit this stop.
- Task< IBusHistoricTimeTable[ ]?> GetArchivedTimeTable (DateTime date)

  Gets the archived real bus departure and arrival times along with their time table history at this specific bus stop.
- bool IsArchivedTimeTableCached (DateTime date)

  Tells you if a file has been cached or not on disk.
- Task< IBusHistoricTimeTable[ ]?> GetWeakArchivedTimeTable (DateTime date)

  Get "Weak" archived timetable doesn't actually call-upon the API feed. It will look at the cached data on disk, finding services that visit the stop, ask for all their timetables and then filter out all the records that are not about this stop. This means that you might end up missing out on some data if you've not cached the service. However it will be significantly faster than actually calling upon the API feed.

Properties

- string AtcoCode `[get]`

  The unique identifier for a bus stop.
- string CommonName `[get]`

  The public, easy to understand stop name.
- string Latitude `[get]`

  The latitude of the bus stop
- string Longitude `[get]`

  The longitude of the bus stop
- string Bearing `[get]`

  The bearing of the bus stop
- string?[ ] Services `[get]`

  A list of the IDs of the services which stop at this stop.

5.19.1 Detailed Description

A class which represents a single bus stop.

5.19.2 Member Function Documentation

5.19.2.1 GetArchivedTimeTable()

```
Task<IBusHistoricTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_API_↩
Interface.IBusStop.GetArchivedTimeTable (
            DateTime date )
```

Gets the archived real bus departure and arrival times along with their time table history at this specific bus stop.

Parameters

date	The date you want time table data for. This should be a date in the past.

Returns

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop.

5.19.2.2 GetServices()

```
IBusService [] Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop.↩
GetServices ( )
```

Finds the 'BusService' object for all of the bus services which visit this stop.

Returns

A list of BusService Objects for services which visit this bus stop.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop.

5.19.2.3 GetWeakArchivedTimeTable()

```
Task<IBusHistoricTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_API_↩
Interface.IBusStop.GetWeakArchivedTimeTable (
            DateTime date )
```

Get "Weak" archived timetable doesn't actually call-upon the API feed. It will look at the cached data on disk, finding services that visit the stop, ask for all their timetables and then filter out all the records that are not about this stop. This means that you might end up missing out on some data if you've not cached the service. However it will be significantly faster than actually calling upon the API feed.

Parameters

date	The date to get stop timetable data from.

Returns

Timetable data for the stop, made up of any cache data about it.

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop.

5.19.2.4 IsArchivedTimeTableCached()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop.IsArchived↩
TimeTableCached (
            DateTime date )
```

Tells you if a file has been cached or not on disk.

Parameters

date	The date for the time table date to search for.

Returns

True if the data is cached on disk

GetArchivedTimeTable(DateTime)

Implemented in Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/IBusStop.cs

5.20 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.IBusTimeTable Interface Reference

A class which represents a single time table record, this is a time and particular bus stop for one service.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTime↩ Table:

Public Member Functions

- bool WeakIsStopSame (IBusTimeTable stop2)

  A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.
- bool WeakIsStopSame (IBusStop stop2)

  Used to say that if given a bus stop object is the stop about this timetable record or not.
- string GetId ()

  Used to get a unique ID value to represent the timetable record.
- bool MatchDirection (Direction direction)

  Used to check if the direction of travel of this record matches the value or not.

Properties

- IBusStop Location `[get]`

  The 'BusStop' object for the stop relating to the time table record..
- long Sequence `[get]`

  What number bus stop is this in the buses route, ie 1, is the first stop to visit.
- Boolean IsOutbound `[get]`

  Is this bus heading inbound or outbound.
- string JourneyCode `[get]`
- string RunningBoard `[get]`
- bool IsTimingPoint `[get]`
- DateTime SchArrivalTime `[get]`

  The scheduled arrival time for the bus.
- DateTime SchDepartureTime `[get]`

  The scheduled departure time for the bus.
- IBusService Service `[get]`

  Gets the related 'IBusService' object relating to the time table record.

5.20.1 Detailed Description

A class which represents a single time table record, this is a time and particular bus stop for one service.

5.20.2 Member Function Documentation

5.20.2.1 GetId()

```
string Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.GetId (
)
```

Used to get a unique ID value to represent the timetable record.

Returns

A value to represent the record.

Implemented in Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, Timetable_Optimisation_Re and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.

5.20.2.2 MatchDirection()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.Match↩
Direction (
            Direction direction )
```

Used to check if the direction of travel of this record matches the value or not.

Parameters

direction	

Returns

Implemented in Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, Timetable_Optimisation_Re
and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.

5.20.2.3 WeakIsStopSame() [1/2]

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.WeakIs↩
StopSame (
            IBusStop stop2 )
```

Used to say that if given a bus stop object is the stop about this timetable record or not.

Parameters

stop2	

Returns

Implemented in Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, Timetable_Optimisation_Re
and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.

5.20.2.4 WeakIsStopSame() [2/2]

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.WeakIs↩
StopSame (
            IBusTimeTable stop2 )
```

A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.

Parameters

stop2	Another time table record to compare against.

Returns

Implemented in Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, Timetable_Optimisation_Re and Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub.

5.20.3 Property Documentation

5.20.3.1 IsTimingPoint

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.Is↩
TimingPoint [get]
```

Is this bus stop a timing point or not.

A timing point is a major bus stop, where the buses is expected to wait if its early and should actually arrive on the scheduled time. All non-timing points times are only estimated scheduled times. A timing point is much more accurate and strict timings. A stop which is a timing point for one service is not necessarily a timing point for another service, hence it is stored here and not in the IBusStop.

5.20.3.2 JourneyCode

```
string Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.↩
JourneyCode [get]
```

A unique value that groups a selection of time table records across different bus stops to show one loop/ cycle of a bus services route.

5.20.3.3 RunningBoard

```
string Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.↩
RunningBoard [get]
```

A running board value, represents a group of journeys that one driver is expected to perform. These are therefore sequential services, driven using the same vehicle.

5.20.3.4 Service

```
IBusService Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.↩
Service [get]
```

Gets the related 'IBusService' object relating to the time table record.

Returns

A 'BusService' object for this time table record.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/IBusTimeTable.cs

5.21 Timetable_Optimisation_Recommendations.Timetable_Simulator.↩ JourneyTime Class Reference

A simplistic class used to help represent the journey time between two stops.

Public Member Functions

- JourneyTime (IBusSolidHistoricTimeTable r1, IBusSolidHistoricTimeTable r2, bool isForwardProp)

  The default journey time constructor, takes in two solid historical records and works out key metrics. It is important the two stops are in order, r1 is the first stop and r2 is the next stop.

Properties

- TimeSpan TravelTime [get]

  The time it takes to travel between the two stops.
- TimeSpan **TimeOfInterest** [get]

5.21.1 Detailed Description

A simplistic class used to help represent the journey time between two stops.

5.21.2 Constructor & Destructor Documentation

5.21.2.1 JourneyTime()

```
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime.JourneyTime (
            IBusSolidHistoricTimeTable r1,
            IBusSolidHistoricTimeTable r2,
            bool isForwardProp )
```

The default journey time constructor, takes in two solid historical records and works out key metrics. It is important the two stops are in order, r1 is the first stop and r2 is the next stop.

Parameters

r1	Timetable record one.
r2	Timetable record two.
isForwardProp	Is forward propagation active, else using backwards.

5.21.3 Property Documentation

5.21.3.1 TravelTime

```
TimeSpan Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime.TravelTime
[get]
```

The time it takes to travel between the two stops.

value> The time of departure at r1 (used for forwards propagation) or time of arrival at r2 (used for backward propagation). Dependent upon the input argument to the constructor.

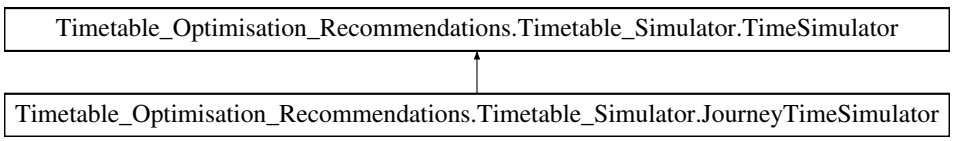
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Simulator/Journey↩ Time.cs

5.22 Timetable_Optimisation_Recommendations.Timetable_Simulator.↩ JourneyTimeSimulator Class Reference

The journey time simulator class takes in a theoretical departure time, two stops to travel between and some other information. It then estimates how long it would likely take to travel between the two stops at this time of day.

Inheritance diagram for Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator:



Public Member Functions

- JourneyTimeSimulator (TimeSpan targetTime, IBusStop start, IBusStop end, IBusService[ ] services, Date↩ Time[ ] dates, bool isForwardProp=true)

  Default constructor for the object, takes in all the required parameters.
- async Task< TimeSpan > ProduceEstimatedTravelTimes (IProgress< double >? progress=null)

  Calculates the estimated journey time, at the specific date and time given between two stops.

Additional Inherited Members

5.22.1 Detailed Description

The journey time simulator class takes in a theoretical departure time, two stops to travel between and some other information. It then estimates how long it would likely take to travel between the two stops at this time of day.

5.22.2 Constructor & Destructor Documentation

5.22.2.1 JourneyTimeSimulator()

```
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator.JourneyTime↩
Simulator (
            TimeSpan targetTime,
            IBusStop start,
            IBusStop end,
            IBusService[] services,
            DateTime[] dates,
            bool isForwardProp = true )
```

Default constructor for the object, takes in all the required parameters.

Parameters

targetTime	The departure or arrival time of interest at the start stop. Dependent upon the direction of propagation.
start	The start stop.
end	The end stop.
services	A list of services that are known to go between the start and end stop consecutively.
dates	A list of dates for when the timetables where the same.
isForwardProp	Is Forward propagating time, default true, else backwards

5.22.3 Member Function Documentation

5.22.3.1 ProduceEstimatedTravelTimes()

```
async Task<TimeSpan> Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime↩
Simulator.ProduceEstimatedTravelTimes (
            IProgress< double >?  progress = null )
```

Calculates the estimated journey time, at the specific date and time given between two stops.

Parameters

progress	Used to feed-back to the GUI the amount of progress made on the simulator.

Returns

The amount of time it would take to journey between the two stops.

If returns 0, there is no data to make an estimate, which would indicate no route goes between these two stops consecutively.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Simulator/Journey↩ TimeSimulator.cs

5.23 Timetable_Optimisation_Recommendations.Timetable_↩ Performance_Evaluator.LatenessRecord Class Reference

Used to represent the lateness of one single record.

Public Member Functions

- LatenessRecord (IBusHistoricTimeTable record)
  The default constructor for the class, takes in a historical record and creates a lateness record out of it.

Properties

- double Lateness `[get]`
  How late the service was
- DateTime **SchArrivalTime** `[get]`
- bool IsLate `[get]`
  Returns true if late, which is earlier than one min or later than five min.

5.23.1 Detailed Description

Used to represent the lateness of one single record.

5.23.2 Constructor & Destructor Documentation

5.23.2.1 LatenessRecord()

```
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord.Lateness↩
Record (
            IBusHistoricTimeTable record )
```

The default constructor for the class, takes in a historical record and creates a lateness record out of it.

Parameters

record	The historical timetable record representing it.

5.23.3 Property Documentation

5.23.3.1 IsLate

```
bool Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord.↩
IsLate [get]
```

Returns true if late, which is earlier than one min or later than five min.

5.23.3.2 Lateness

```
double Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord.↩
Lateness [get]
```

How late the service was

value>The time of day it was meant to have arrived, to see if lateness changes throughout the day.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Performance Evaluator/LatenessRecord.cs

5.24 Timetable_Optimisation_Recommendations.Timetable_↩ Performance_Evaluator.LatenessReport Struct Reference

Used for MVVM to bind to the GUI, this contains data summerising the performance of a service.

Properties

- IBusService Service [get]

  The Service the report pertains to.
- double OnTimePercentage [get]

  value>The average lateness of a service in min.
- string AvgLatenessString [get]

  value>The on time percentage formatted nicely as a string value.
- string **OnTimePercentageString** [get]

5.24.1 Detailed Description

Used for MVVM to bind to the GUI, this contains data summerising the performance of a service.

5.24.2 Property Documentation

5.24.2.1 Service

```
IBusService Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.Lateness↩
Report.Service  [get]
```

The Service the report pertains to.

value>The On Time Percentage as a double.

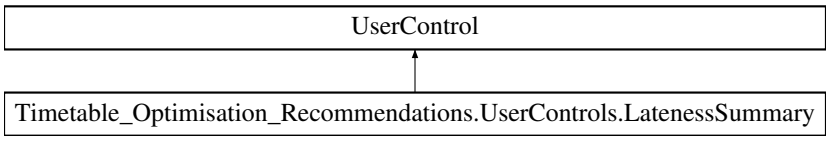
The documentation for this struct was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Performance Evaluator/LatenessReport.cs

5.25 Timetable_Optimisation_Recommendations.UserControls.↩ LatenessSummary Class Reference

The lateness summary report card, shows the performance of the service historically.

Inheritance diagram for Timetable_Optimisation_Recommendations.UserControls.LatenessSummary:



5.25.1 Detailed Description

The lateness summary report card, shows the performance of the service historically.

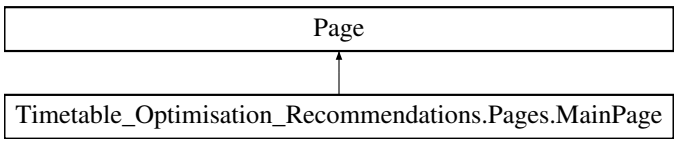
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/UserControls/Lateness↩ Summary.xaml.cs

5.26 Timetable_Optimisation_Recommendations.Pages.MainPage Class Reference

The main start page for the program, this is where the user will select the primary service that they wish to optimize for.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.MainPage:



Public Member Functions

- MainPage ()

  The default program entry point constructor.

Properties

- ObservableCollection< IBusService > ServiceCardCollection = new(BusOperatorFactory.Instance.↩ Operator.GetServices()) `[get]`

  Stores a list of bus services that the operator operates.

5.26.1 Detailed Description

The main start page for the program, this is where the user will select the primary service that they wish to optimize for.

5.26.2 Constructor & Destructor Documentation

5.26.2.1 MainPage()

```
Timetable_Optimisation_Recommendations.Pages.MainPage.MainPage ( )
```

The default program entry point constructor.

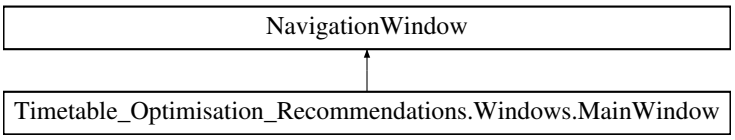
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/MainPage.↩ xaml.cs

5.27 Timetable_Optimisation_Recommendations.Windows.MainWindow Class Reference

The main entry point for the application, but the logic is contained within the page.

Inheritance diagram for Timetable_Optimisation_Recommendations.Windows.MainWindow:



5.27.1 Detailed Description

The main entry point for the application, but the logic is contained within the page.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Windows/MainWindow.xaml.cs

5.28 Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move Struct Reference

Used to represent a single move of the search algorithm, this involves making a change to one services timetable, on one running-board. A single timetable record is moved and then the surrounding records will also need to be edited in forwards and backwards propagation.

Public Member Functions

- override string ToString ()

  Provides a string representation of the move. This is only accurate if TargetRecord.SetSuggestedToReal() hasn't already been called.

Properties

- IBusService Service `[get]`

  States what service this moves alters.
- BlamedBusTimeTable[ ] TimeTable `[get]`

  value>The IDs of the records in the timetable that have actually changed.
- List< string > ChangedRecordsIDs `[get]`

  value>The total amount of minuets changes in the move from the initial solution.
- double ChangeAmount `[get]`

  value>The new proposed scheduled arrival time.
- DateTime ProposedSchArrivalTime `[get]`

  value>The new proposed scheduled departure time.
- DateTime ProposedSchDepartureTime `[get]`

  value>The timetable record highlighted as being the problem.
- BlamedBusTimeTable **TargetRecord** `[get]`

5.28.1 Detailed Description

Used to represent a single move of the search algorithm, this involves making a change to one services timetable, on one running-board. A single timetable record is moved and then the surrounding records will also need to be edited in forwards and backwards propagation.

As such a move is represented as the service it is about and an array of blamed timetable records, which contain the new timetable for the service. Most records won't have actually moved.

5.28.2 Member Function Documentation

5.28.2.1 ToString()

```
override string Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move.To↩
String ( )
```

Provides a string representation of the move. This is only accurate if TargetRecord.SetSuggestedToReal() hasn't already been called.

Returns

A string representation of the changes.

5.28.3 Property Documentation

5.28.3.1 Service

```
IBusService Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move.Service
[get]
```

States what service this moves alters.

value>The new timetable for the service after the move has been applied.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Search Algorithm/↩ Tabu Search/Move.cs

5.29 Timetable_Optimisation_Recommendations.Search_Algorithm.↩ Tabu_Search.MoveSelector Class Reference

The move-selector is used to select the best move out of an array of moves, the neighborhood, to generate a new solution.

Public Member Functions

- MoveSelector (TimeTableEvaluator evaluator)

  Default constructor, takes in a reference to the evaluator object.
- async Task<(Solution, Move)> BestMoveSelectorAsync (Move[] moves, TabuList tabuList, IProgress< AdvancedProgressReporting >? progress=null)

  Given an array of moves, identify which is the best move and return the new solution with that move applied to it.

5.29.1 Detailed Description

The move-selector is used to select the best move out of an array of moves, the neighborhood, to generate a new solution.

5.29.2 Constructor & Destructor Documentation

5.29.2.1 MoveSelector()

```
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.MoveSelector.MoveSelector
(
            TimeTableEvaluator evaluator )
```

Default constructor, takes in a reference to the evaluator object.

Parameters

evaluator	

5.29.3 Member Function Documentation

5.29.3.1 BestMoveSelectorAsync()

```
async Task<(Solution, Move)> Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_↩
Search.MoveSelector.BestMoveSelectorAsync (
            Move[] moves,
            TabuList tabuList,
            IProgress< AdvancedProgressReporting >?  progress = null )
```

Given an array of moves, identify which is the best move and return the new solution with that move applied to it.

Parameters

moves	An array of possible moves to make.
progress	Used to report back to the GUI or listener of the progress of this task.

Returns

The best solution which is the best move applied to the current solution. Along with the selected move that got us there.

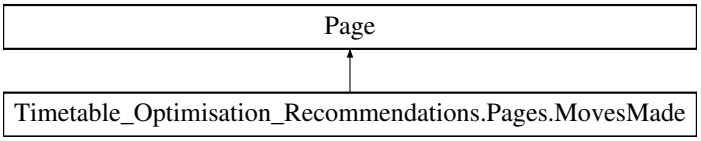
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Search Algorithm/↩ Tabu Search/MoveSelector.cs

5.30 Timetable_Optimisation_Recommendations.Pages.MovesMade Class Reference

Used to display the moves that have been made by the search algorithm after it has completed and displays the finalized timetables.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.MovesMade:



Public Member Functions

- MovesMade (Solution startSolution, Solution solution, int iterationOfBestMove, List<(Move moves, int score)> movesMade)

  The default constructor for the results page.

5.30.1 Detailed Description

Used to display the moves that have been made by the search algorithm after it has completed and displays the finalized timetables.

5.30.2 Constructor & Destructor Documentation

5.30.2.1 MovesMade()

```
Timetable_Optimisation_Recommendations.Pages.MovesMade.MovesMade (
            Solution startSolution,
            Solution solution,
            int iterationOfBestMove,
            List<(Move moves, int score)> movesMade )
```

The default constructor for the results page.

Parameters

startSolution	The original input solution.
solution	The best solution found.
movesMade	A list of moves that were made.
iterationOfBestMove	The iteration count where the best move was found.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/MovesMade.↩ xaml.cs

5.31 Timetable_Optimisation_Recommendations.Search_Algorithm.↩ Tabu_Search.NeighbourhoodSolution Class Reference

Used to generate the neighbourhood of solutions for the search algorithm.

Public Member Functions

- NeighbourhoodSolution (TimeTableEvaluator evaluator, int? neighbourhoodSize=null, int? candidateList↩ Size=null)

  The default constructor for the NeighbourhoodSolution class.
- async Task< Move[]> GenerateNegibourhood (TabuList tabuList, IProgress< AdvancedProgressReporting >? progress=null)

  Generates the negibourhood of solutions, returns an array of tuples of solutions, the solution only contains changes to one service, it is assumed that all other services results won't have changed. This is done for efficiency purposes, to both save memory and because you don't need to recalculate objective function value on everything.

5.31.1 Detailed Description

Used to generate the neighbourhood of solutions for the search algorithm.

5.31.2 Constructor & Destructor Documentation

5.31.2.1 NeighbourhoodSolution()

```
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.NeighbourhoodSolution.↩
NeighbourhoodSolution (
            TimeTableEvaluator evaluator,
            int?  neighbourhoodSize = null,
            int?  candidateListSize = null )
```

The default constructor for the NeighbourhoodSolution class.

Parameters

evaluator	A reference to the main evaluator.
neighbourhoodSize	The size of the neighborhood to generate, only needed if not default, should be greater than 1, preferably bigger..
candidateListSize	The size of the candidate list to generate, only needed if not default, should be less than or equal to neighborhood.

5.31.3 Member Function Documentation

5.31.3.1 GenerateNegibourhood()

```
async Task<Move[]> Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.↩
NeighbourhoodSolution.GenerateNegibourhood (
            TabuList tabuList,
            IProgress< AdvancedProgressReporting >?  progress = null )
```

Generates the negibourhood of solutions, returns an array of tuples of solutions, the solution only contains changes to one service, it is assumed that all other services results won't have changed. This is done for efficiency purposes, to both save memory and because you don't need to recalculate objective function value on everything.

Returns

An array of solutions, one per candidate list.

It is theoretically possible, but very unlikely that this would return back zero moves. But it can do if a lot of the search space is tabu and there is only a small search space. Regardless this should be checked for.

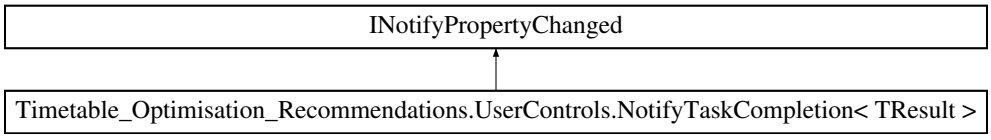
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Search Algorithm/↩ Tabu Search/NeighbourhoodSolution.cs

5.32 Timetable_Optimisation_Recommendations.UserControls.Notify↩ TaskCompletion< TResult > Class Template Reference

Helper class to wrap around a Task to provide more information usable for UI databinding scenarios. As discussed in MSDN Magazine: https://msdn.microsoft.com/magazine/dn605875.

Inheritance diagram for Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >:



Public Member Functions

- NotifyTaskCompletion (Task< TResult > task)

  Initializes a new instance of the NotifyTaskCompletion< TResult > class.

Properties

- Task< TResult > Task [get]

  Gets the task that is being waited on.
- Task TaskCompletion [get]

  Gets the task wrapper task.
- TResult?? Result [get]

  Gets the result of the given task.
- TaskStatus Status [get]

  Gets the status of the task.
- bool IsCompleted [get]

  Gets a value indicating whether the task is completed.
- bool IsNotCompleted [get]

  Gets a value indicating whether the task is not completed.
- bool IsSuccessfullyCompleted [get]

  Gets a value indicating whether the task was successfully completed.
- bool IsCanceled [get]

  Gets a value indicating whether the task was canceled.
- bool IsFaulted [get]

  Gets a value indicating whether there was an error with the task.
- AggregateException Exception [get]

  Gets the exception which occured on the task (if one occurred).
- Exception InnerException [get]

  Gets the inner exception of the task.
- string ErrorMessage [get]

  Gets the error message of the task.

Events

- PropertyChangedEventHandler PropertyChanged

  PropertyChanged event.

5.32.1 Detailed Description

Helper class to wrap around a Task to provide more information usable for UI databinding scenarios. As discussed in MSDN Magazine: https://msdn.microsoft.com/magazine/dn605875.

Template Parameters

TResult	Type of result returned by task.

5.32.2 Constructor & Destructor Documentation

5.32.2.1 NotifyTaskCompletion()

```
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.NotifyTaskCompletion
(
            Task< TResult > task )
```

Initializes a new instance of the NotifyTaskCompletion<TResult> class.

Parameters

task	Task to wait on.

5.32.3 Property Documentation

5.32.3.1 ErrorMessage

```
string Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Error↩
Message  [get]
```

Gets the error message of the task.

5.32.3.2 Exception

```
AggregateException Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion<
TResult >.Exception  [get]
```

Gets the exception which occured on the task (if one occurred).

5.32.3.3 InnerException

```
Exception Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult
>.InnerException  [get]
```

Gets the inner exception of the task.

5.32.3.4 IsCanceled

```
bool Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Is↩
Canceled [get]
```

Gets a value indicating whether the task was canceled.

5.32.3.5 IsCompleted

```
bool Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Is↩
Completed [get]
```

Gets a value indicating whether the task is completed.

5.32.3.6 IsFaulted

```
bool Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Is↩
Faulted [get]
```

Gets a value indicating whether there was an error with the task.

5.32.3.7 IsNotCompleted

```
bool Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Is↩
NotCompleted [get]
```

Gets a value indicating whether the task is not completed.

5.32.3.8 IsSuccessfullyCompleted

```
bool Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Is↩
SuccessfullyCompleted [get]
```

Gets a value indicating whether the task was successfully completed.

5.32.3.9 Result

```
TResult?? Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult
>.Result [get]
```

Gets the result of the given task.

5.32.3.10 Status

```
TaskStatus Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult
>.Status [get]
```

Gets the status of the task.

5.32.3.11 Task

```
Task<TResult> Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult
>.Task [get]
```

Gets the task that is being waited on.

5.32.3.12 TaskCompletion

```
Task Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Task↩
Completion [get]
```

Gets the task wrapper task.

5.32.4 Event Documentation

5.32.4.1 PropertyChanged

```
PropertyChangedEventHandler Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion<
TResult >.PropertyChanged
```

PropertyChanged event.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/UserControls/Notify↩ TaskCompletion.cs

5.33 Timetable_Optimisation_Recommendations.Timetable_↩ Performance_Evaluator.PerformanceEvaluator Class Reference

A class used to generate the performance of the historical/current timetables.

Public Member Functions

- void AddRecords (IBusService service, IBusHistoricTimeTable[]? records)

  Used to add onto a services lateness record.
- void GenerateLatenessReport ()

  Generates the summarized report for each service, this is what is being displayed to the end user.

Properties

- List< LatenessReport > **ServiceLatenessReports** = new() `[get]`

5.33.1 Detailed Description

A class used to generate the performance of the historical/current timetables.

5.33.2 Member Function Documentation

5.33.2.1 AddRecords()

```
void Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.Performance↩
Evaluator.AddRecords (
            IBusService service,
            IBusHistoricTimeTable?[] records )
```

Used to add onto a services lateness record.

Parameters

service	The service you are adding to.
records	The records to convert and add.

5.33.2.2 GenerateLatenessReport()

```
void Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.Performance↩
Evaluator.GenerateLatenessReport ( )
```

Generates the summarized report for each service, this is what is being displayed to the end user.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Performance Evaluator/PerformanceEvaluator.cs

5.34 Timetable_Optimisation_Recommendations.Timetable_Evaluator.↩ PreEvaluatorChecks Class Reference

Pre-evaluator checks is run before actually running the real evaluator and is used to download all of the data that is required and evaluate the performance of the old timetable.

Public Member Functions

- PreEvaluatorChecks (Cluster cluster, RouteSegmentCollection segmentCollection)

  The default constructor for the class, takes in the objects created from the previous steps.
- TimeTableEvaluator EvaluateTimeTable ()

  Produces the evaluator object from the pre-evaluator.
- async Task DownloadAllFilesNeeded (IProgress< AdvancedProgressReporting >? progress)

  Does the work, downloads all of the data needed so that it is in tire 2 and 1 cache. Also works out the performance metrics of this data.

Properties

- PerformanceEvaluator PerformanceEvaluator = new() `[get]`

  value>Keeps track of how many tasks to complete.

5.34.1 Detailed Description

Pre-evaluator checks is run before actually running the real evaluator and is used to download all of the data that is required and evaluate the performance of the old timetable.

5.34.2 Constructor & Destructor Documentation

5.34.2.1 PreEvaluatorChecks()

```
Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluatorChecks.PreEvaluator↩
Checks (
            Cluster cluster,
            RouteSegmentCollection segmentCollection )
```

The default constructor for the class, takes in the objects created from the previous steps.

Parameters

cluster	Stores information about the dates to request data for.
segmentCollection	Stores information about the route segments.

5.34.3 Member Function Documentation

5.34.3.1 DownloadAllFilesNeeded()

```
async Task Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluatorChecks.↩
DownloadAllFilesNeeded (
            IProgress< AdvancedProgressReporting >?  progress )
```

Does the work, downloads all of the data needed so that it is in tire 2 and 1 cache. Also works out the performance metrics of this data.

Parameters

progress	Used to report back the progress to the GUI.

Returns

Caches all of the required data for the search.

5.34.3.2 EvaluateTimeTable()

```
TimeTableEvaluator Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluator↩
Checks.EvaluateTimeTable ( )
```

Produces the evaluator object from the pre-evaluator.

Returns

The final evaluator to actually perform the search

You shouldn't call this before DownloadAllFilesNeeded has been called and completed.

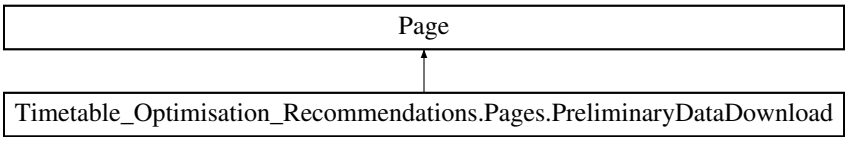
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Evaluator/Pre↩ EvaluatorChecks.cs

5.35 Timetable_Optimisation_Recommendations.Pages.Preliminary↩ DataDownload Class Reference

The preliminary data download lets you download all of the data files that you are going to need for the search. It also lets you see the services past performance.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload:



Public Member Functions

- PreliminaryDataDownload (Cluster cluster, RouteSegmentCollection collection)

  The default constructor, takes in the information from the previous pages.

Properties

- AdvancedProgressReporting **Reporter** = new() `[get]`

5.35.1 Detailed Description

The preliminary data download lets you download all of the data files that you are going to need for the search. It also lets you see the services past performance.

5.35.2 Constructor & Destructor Documentation

5.35.2.1 PreliminaryDataDownload()

```
Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload.PreliminaryDataDownload (
            Cluster cluster,
            RouteSegmentCollection collection )
```

The default constructor, takes in the information from the previous pages.

Parameters

cluster	The dates that the user is requesting data for.
collection	All of the shared route-segments containing information on other services we also want data for.

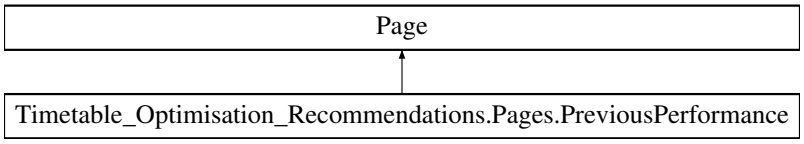
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/Preliminary↩ DataDownload.xaml.cs

5.36 Timetable_Optimisation_Recommendations.Pages.Previous↩ Performance Class Reference

Once all of the data has been downloaded in the pre-evaluator checks show the performance metrics to the user. This is the final stage before starting the actual search.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.PreviousPerformance:



Public Member Functions

- PreviousPerformance (PreEvaluatorChecks preEvaluator)

  The default constructor to the page, takes in the pre-evaluator results

Properties

- ObservableCollection< LatenessReport > ServiceCardCollection = new() `[get]`

  Used to store the list of services and their performance metrics.

5.36.1 Detailed Description

Once all of the data has been downloaded in the pre-evaluator checks show the performance metrics to the user. This is the final stage before starting the actual search.

5.36.2 Constructor & Destructor Documentation

5.36.2.1 PreviousPerformance()

```
Timetable_Optimisation_Recommendations.Pages.PreviousPerformance.PreviousPerformance (
            PreEvaluatorChecks preEvaluator )
```

The default constructor to the page, takes in the pre-evaluator results

Parameters

preEvaluator	The pre-evaluator results from the previous page.

5.36.3 Property Documentation

5.36.3.1 ServiceCardCollection

```
ObservableCollection<LatenessReport> Timetable_Optimisation_Recommendations.Pages.Previous↩
Performance.ServiceCardCollection = new()  [get]
```

Used to store the list of services and their performance metrics.

value>Used to store the results of the pre-evaluator to give it to the evaluator next.

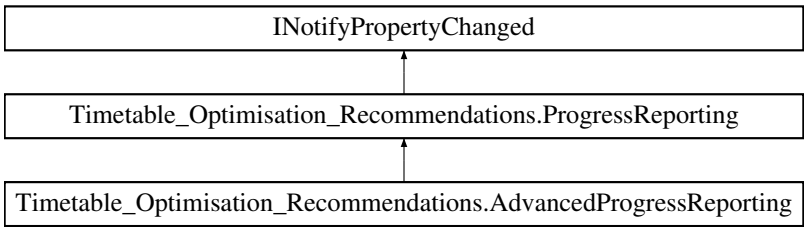
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/Previous↩ Performance.xaml.cs

5.37 Timetable_Optimisation_Recommendations.ProgressReporting Class Reference

Used to report back the progress of th task to the GUI.

Inheritance diagram for Timetable_Optimisation_Recommendations.ProgressReporting:



Public Member Functions

- ProgressReporting ()

  The default constructor, sets everything to be equal to zero.
- ProgressReporting (double value, string message)

  A constructor used to specify a tasks progress and a message for the task.
- void Update (ProgressReporting reporter)

  Used to update a progress reporter with information from another one.
- void Clear ()

  Clears all values back to zero/null.

Protected Member Functions

- void OnPropertyChanged (string? name=null)

  Used to tell the GUI to update whenever a value in th class is changed.

Properties

- double Value `[get, set]`

  value>The message of the whole task, saying everything that has been completed.
- string Message `[get, set]`

  value>Defines if a GUI element should be visible or not. Only shows the thing while progress is not completed.
- Visibility? **Visibility** `[get]`

Events

- PropertyChangedEventHandler? PropertyChanged

  Used to tell the GUI to refresh/update to the actual value.

5.37.1 Detailed Description

Used to report back the progress of th task to the GUI.

5.37.2 Constructor & Destructor Documentation

5.37.2.1 ProgressReporting() [1/2]

```
Timetable_Optimisation_Recommendations.ProgressReporting.ProgressReporting ( )
```

The default constructor, sets everything to be equal to zero.

5.37.2.2 ProgressReporting() [2/2]

```
Timetable_Optimisation_Recommendations.ProgressReporting.ProgressReporting (
            double value,
            string message )
```

A constructor used to specify a tasks progress and a message for the task.

Parameters

value	The overall progress for the task.
message	The message of the last completed task.

5.37.3 Member Function Documentation

5.37.3.1 Clear()

```
void Timetable_Optimisation_Recommendations.ProgressReporting.Clear ( )
```

Clears all values back to zero/null.

5.37.3.2 OnPropertyChanged()

```
void Timetable_Optimisation_Recommendations.ProgressReporting.OnPropertyChanged (
            string?  name = null )  [protected]
```

Used to tell the GUI to update whenever a value in th class is changed.

Parameters

name	

5.37.3.3 Update()

```
void Timetable_Optimisation_Recommendations.ProgressReporting.Update (
            ProgressReporting reporter )
```

Used to update a progress reporter with information from another one.

Parameters

reporter	The progress reporter to take in feedback from.

5.37.4 Event Documentation

5.37.4.1 PropertyChanged

```
PropertyChangedEventHandler?  Timetable_Optimisation_Recommendations.ProgressReporting.↩
PropertyChanged
```

Used to tell the GUI to refresh/update to the actual value.

value>The value of the task as a whole.

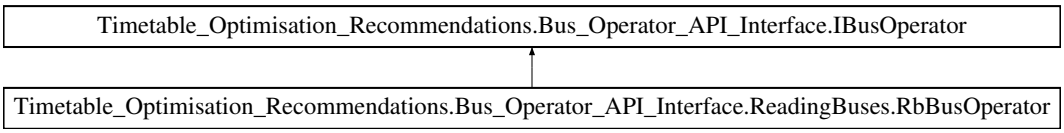
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/ProgressReporting.cs

5.38 Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator Class Reference

An Implementation of the IBusOperator interface for the Reading Buses API.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator:



Public Member Functions

- bool IsService (string serviceNumber)

  Checks to see if a service of that number exists or not in the API feed.
- IBusStop GetLocation (string atcoCode)

  Get a bus stop location based upon a bus stops location code
- bool IsLocation (string atcoCode)

  Checks to see if the acto code for the bus stop exists in the API feed or not.
- void InvalidateCache ()

  Deletes any Cache data stored, use this only if you need to force new data cache.
- IBusService GetService (string serviceNumber)

  Returns a service which matches the Service Number passed.
- void ForceUpdateCache ()

  Forces the current data stored in the bus operator object to be saved into Cache. You would need to do this if you've made some lazy API requests.
- IBusService[ ] GetServices ()

  Gets an array of all the IBusServices Objects.

Static Public Member Functions

- static async Task< IBusOperator > Initialise (string apiKey)

  Used to initially initialise the ReadingBuses Object, it is recommended you do this in your programs start up.
- static IBusOperator **GetInstance** ()

Static Public Attributes

- static readonly string **CacheDirectory** = "cache/ReadingBuses"

Properties

- ConcurrentDictionary< string, RbBusStop > **Locations** = new() `[get]`
- ConcurrentDictionary< string, RbBusService > **Services** = new() `[get]`

5.38.1 Detailed Description

An Implementation of the IBusOperator interface for the Reading Buses API.

5.38.2 Member Function Documentation

5.38.2.1 ForceUpdateCache()

```
void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Operator.ForceUpdateCache ( )
```

Forces the current data stored in the bus operator object to be saved into Cache. You would need to do this if you've made some lazy API requests.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.38.2.2 GetLocation()

```
IBusStop Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.Rb↩
BusOperator.GetLocation (
            string atcoCode )
```

Get a bus stop location based upon a bus stops location code

Parameters

atcoCode	The code of the bus stop

Returns

A Bus Stop object for the Atco Code specified.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.38.2.3 GetService()

```
IBusService Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.↩
RbBusOperator.GetService (
            string serviceNumber )
```

Returns a service which matches the Service Number passed.

Parameters

serviceNumber	The service number/ID for the service you wish to be returned eg: 17 or 22.

Returns

The services matching the ID.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.38.2.4 GetServices()

```
IBusService [] Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩
Buses.RbBusOperator.GetServices ( )
```

Gets an array of all the IBusServices Objects.

Returns

An array of all the bus services.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.38.2.5 Initialise()

```
static async Task<IBusOperator> Timetable_Optimisation_Recommendations.Bus_Operator_API_↩
Interface.ReadingBuses.RbBusOperator.Initialise (
            string apiKey )  [static]
```

Used to initially initialise the ReadingBuses Object, it is recommended you do this in your programs start up.

Parameters

apiKey	The Reading Buses API Key, get your own from http://rtl2.ods-live.co.uk/cms/apiservice

Returns

An instance of the library controller. This same instance can be got by calling the "GetInstance" method.

Exceptions

ReadingBusesApiExceptionBadQuery	Can throw an exception if you pass an invalid or expired API Key.

See Bus_Operator_API_Interface.ReadingBuses.GetInstance() to get any future instances afterwards.

5.38.2.6 InvalidateCache()

```
void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Operator.InvalidateCache ( )
```

Deletes any Cache data stored, use this only if you need to force new data cache.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.38.2.7 IsLocation()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Operator.IsLocation (
            string atcoCode )
```

Checks to see if the acto code for the bus stop exists in the API feed or not.

Parameters

atcoCode	The ID Code for a bus stop.

Returns

True or False depending on if the stop is in the API feed or not.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.38.2.8 IsService()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Operator.IsService (
            string serviceNumber )
```

Checks to see if a service of that number exists or not in the API feed.

Parameters

serviceNumber	The service number to find.

Returns

True or False for if a service is the API feed or not.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

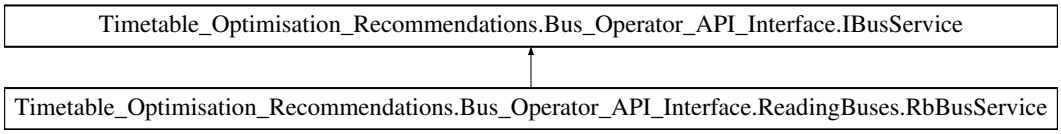
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/ReadingBuses/RB_BusOperator.cs

5.39 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.ReadingBuses.RbBusService Class Reference

An Implementation of the IBusService interface for the Reading Buses API.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩ Buses.RbBusService:



Public Member Functions

- async Task< IBusStop[ ]> GetLocations (Direction direction=Direction.Both)

  Gets an array of 'BusStop' objects the bus service travels too as an array of BusStop objects. If the API is invalid and links to a Bus Stop not in the list of locations it will simply be ignored.
- bool IsArchivedTimeTableCached (DateTime date)

  Tells you if a file has been cached or not on disk.
- bool IsWeakServiceSame (IBusService service)

  Given another IBusService Object, check if it is the same service or not. Only by comparing the service ID value.
- async Task< IBusHistoricTimeTable[ ]?> GetArchivedTimeTable (DateTime date)

  Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date.
- bool IsTimeTableCached (DateTime date)

  Tells you if a file has been cached or not on disk.
- async Task< IBusTimeTable[ ]?> GetTimeTable (DateTime date)

  Gets the planned timetable departure and arrival times for this service on a specific date.
- override string **ToString** ()
- override int **GetHashCode** ()
- override bool **Equals** (object? obj)
- bool **Equals** (RbBusService? other)

Static Public Member Functions

- static **operator RbBusService** (BusService service)

Properties

- string **ServiceId** = string.Empty `[get]`

5.39.1 Detailed Description

An Implementation of the IBusService interface for the Reading Buses API.

5.39.2 Member Function Documentation

5.39.2.1 GetArchivedTimeTable()

```
async Task<IBusHistoricTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_↩
API_Interface.ReadingBuses.RbBusService.GetArchivedTimeTable (
            DateTime date )
```

Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date.

Parameters

date	the date on which you want a archived timetable data for. This should be a date in the past.

Returns

An array of time table records, containing the scheduled and actual arrival and departure times of buses.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.

5.39.2.2 GetLocations()

```
async Task<IBusStop[]> Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.↩
ReadingBuses.RbBusService.GetLocations (
            Direction direction = Direction.Both )
```

Gets an array of 'BusStop' objects the bus service travels too as an array of BusStop objects. If the API is invalid and links to a Bus Stop not in the list of locations it will simply be ignored.

Parameters

direction	Used to filter by the direction of travel the stops are on.

Returns

An array of BusStop objects for the stops visited by this service.

It is assumed that the ordering of the array is the ordering in which a service will visit all of the stops. If the ordering is incorrect the route-segment finder will fail.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.

5.39.2.3 GetTimeTable()

```
async Task<IBusTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.↩
ReadingBuses.RbBusService.GetTimeTable (
            DateTime date )
```

Gets the planned timetable departure and arrival times for this service on a specific date.

Parameters

date	the date on which you want a archived timetable data for. This should be a date in the past.

Returns

An array of time table records, containing the planned scheduled and actual arrival and departure times of buses.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.

5.39.2.4 IsArchivedTimeTableCached()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Service.IsArchivedTimeTableCached (
            DateTime date )
```

Tells you if a file has been cached or not on disk.

Parameters

date	The date for the time table date to search for.

Returns

True if the data is cached on disk

GetArchivedTimeTable(DateTime)

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.

5.39.2.5 IsTimeTableCached()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Service.IsTimeTableCached (
            DateTime date )
```

Tells you if a file has been cached or not on disk.

Parameters

date	The date for the time table date to search for.

Returns

True if the data is cached on disk

GetTimeTable(DateTime)

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.

5.39.2.6 IsWeakServiceSame()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Service.IsWeakServiceSame (
            IBusService service )
```

Given another IBusService Object, check if it is the same service or not. Only by comparing the service ID value.

Parameters

service	The other service you wish to compare against.

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService.

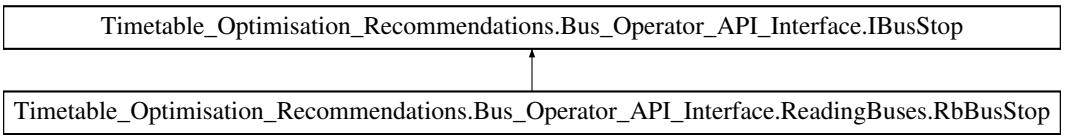
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/ReadingBuses/RB_BusService.cs

5.40 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.ReadingBuses.RbBusStop Class Reference

An Implementation of the IBusStop interface for the Reading Buses API.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩ Buses.RbBusStop:



Public Member Functions

- IBusService[ ] GetServices ()

  Finds the 'BusService' object for all of the bus services which visit this stop.
- bool IsArchivedTimeTableCached (DateTime date)

  Tells you if a file has been cached or not on disk.
- async Task< IBusHistoricTimeTable[ ]?> GetArchivedTimeTable (DateTime date)

  Gets the archived real bus departure and arrival times along with their time table history at this specific bus stop.
- async Task< IBusHistoricTimeTable[ ]?> GetWeakArchivedTimeTable (DateTime date)

  Get "Weak" archived timetable doesn't actually call-upon the API feed. It will look at the cached data on disk, finding services that visit the stop, ask for all their timetables and then filter out all the records that are not about this stop. This means that you might end up missing out on some data if you've not cached the service. However it will be significantly faster than actually calling upon the API feed.
- override bool **Equals** (object? obj)
- bool **Equals** (RbBusStop? other)
- override int **GetHashCode** ()
- override string **ToString** ()

Static Public Member Functions

- static **operator RbBusStop** (BusStop stop)

Properties

- string **AtcoCode** = string.Empty `[get]`
- string **CommonName** = string.Empty `[get]`
- string **Latitude** = string.Empty `[get]`
- string **Longitude** = string.Empty `[get]`
- string **Bearing** = string.Empty `[get]`
- string?[ ] **Services** `[get]`

5.40.1 Detailed Description

An Implementation of the IBusStop interface for the Reading Buses API.

5.40.2 Member Function Documentation

5.40.2.1 GetArchivedTimeTable()

```
async Task<IBusHistoricTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_↩
API_Interface.ReadingBuses.RbBusStop.GetArchivedTimeTable (
            DateTime date )
```

Gets the archived real bus departure and arrival times along with their time table history at this specific bus stop.

Parameters

date	The date you want time table data for. This should be a date in the past.

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop.

5.40.2.2 GetServices()

```
IBusService [] Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩
Buses.RbBusStop.GetServices ( )
```

Finds the 'BusService' object for all of the bus services which visit this stop.

Returns

A list of BusService Objects for services which visit this bus stop.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop.

5.40.2.3 GetWeakArchivedTimeTable()

```
async Task<IBusHistoricTimeTable[]?> Timetable_Optimisation_Recommendations.Bus_Operator_↩
API_Interface.ReadingBuses.RbBusStop.GetWeakArchivedTimeTable (
            DateTime date )
```

Get "Weak" archived timetable doesn't actually call-upon the API feed. It will look at the cached data on disk, finding services that visit the stop, ask for all their timetables and then filter out all the records that are not about this stop. This means that you might end up missing out on some data if you've not cached the service. However it will be significantly faster than actually calling upon the API feed.

Parameters

date	The date to get stop timetable data from.

Returns

Timetable data for the stop, made up of any cache data about it.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop.

5.40.2.4 IsArchivedTimeTableCached()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBus↩
Stop.IsArchivedTimeTableCached (
            DateTime date )
```

Tells you if a file has been cached or not on disk.

Parameters

date	The date for the time table date to search for.

Returns

True if the data is cached on disk

GetArchivedTimeTable(DateTime)

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop.

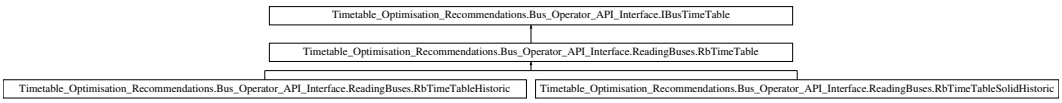
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/ReadingBuses/RB_BusStop.cs

5.41 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.ReadingBuses.RbTimeTable Class Reference

An Implementation of the IBusTimetable interface for the Reading Buses API.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩ Buses.RbTimeTable:



Public Member Functions

- bool WeakIsStopSame (IBusTimeTable stop2)

  A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.
- string GetId ()

  Used to get a unique ID value to represent the timetable record.
- bool WeakIsStopSame (IBusStop stop2)

  Used to say that if given a bus stop object is the stop about this timetable record or not.
- bool MatchDirection (Direction direction)

  Used to check if the direction of travel of this record matches the value or not.

Static Public Member Functions

- static **operator RbTimeTable** (BusTimeTable inputTimeTable)

Protected Attributes

- string **ServiceId** = string.Empty
- string **AtcoCode** = string.Empty

Properties

- long **Sequence** `[get, protected set]`
- bool **IsOutbound** `[get, protected set]`
- string **JourneyCode** = string.Empty `[get, protected set]`
- bool **IsTimingPoint** `[get, protected set]`
- DateTime **SchArrivalTime** `[get, protected set]`
- DateTime **SchDepartureTime** `[get, protected set]`
- string **RunningBoard** = string.Empty `[get, protected set]`
- IBusService **Service** `[get, protected set]`
- IBusStop **Location** `[get, protected set]`

5.41.1 Detailed Description

An Implementation of the IBusTimetable interface for the Reading Buses API.

5.41.2 Member Function Documentation

5.41.2.1 GetId()

```
string Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩
Table.GetId ( )
```

Used to get a unique ID value to represent the timetable record.

Returns

A value to represent the record.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.41.2.2 MatchDirection()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩
Table.MatchDirection (
            Direction direction )
```

Used to check if the direction of travel of this record matches the value or not.

Parameters

direction	

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.41.2.3 WeakIsStopSame() [1/2]

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩
Table.WeakIsStopSame (
            IBusStop stop2 )
```

Used to say that if given a bus stop object is the stop about this timetable record or not.

Parameters

stop2	

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

5.41.2.4 WeakIsStopSame() [2/2]

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩
Table.WeakIsStopSame (
            IBusTimeTable stop2 )
```

A faster way to compare if two IBusTimeTable records are about the same bus stop. By simply comparing their string atco code, as opposed to finding and comparing the two bus stop objects.

Parameters

stop2	Another time table record to compare against.

Returns

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable.

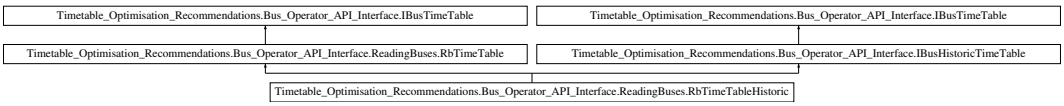
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/ReadingBuses/RB_TimeTable.cs

5.42 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.ReadingBuses.RbTimeTableHistoric Class Reference

An Implementation of the IBusHistoricTimetable interface for the Reading Buses API.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩ Buses.RbTimeTableHistoric:



Public Member Functions

- bool CouldBeSolid ()

  Returns if the record could be made a "solid" record or not. A solid record is one with reported arrival and departure times.
- IBusSolidHistoricTimeTable GetSolid ()

  Gets the solid representation of the same object.

Static Public Member Functions

- static **operator RbTimeTableHistoric** (ArchivedBusTimeTable inputTimeTable)

Properties

- DateTime? **ActArrivalTime** `[get]`
- DateTime? **ActDepartureTime** `[get]`

Additional Inherited Members

5.42.1 Detailed Description

An Implementation of the IBusHistoricTimetable interface for the Reading Buses API.

5.42.2 Member Function Documentation

5.42.2.1 CouldBeSolid()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩
TableHistoric.CouldBeSolid ( )
```

Returns if the record could be made a "solid" record or not. A solid record is one with reported arrival and departure times.

Returns

true if Actual Arrival and Departure have values.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoricTimeTable.

5.42.2.2 GetSolid()

```
IBusSolidHistoricTimeTable Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.↩
ReadingBuses.RbTimeTableHistoric.GetSolid ( )
```

Gets the solid representation of the same object.

Returns

Gets the solid equivalence object.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusHistoricTimeTable.

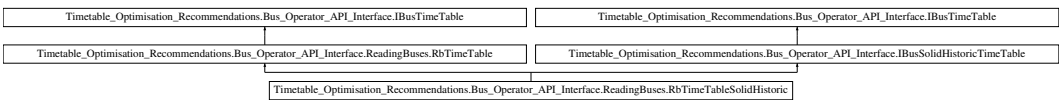
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/ReadingBuses/RB_TimeTableHistoric.cs

5.43 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.ReadingBuses.RbTimeTableSolidHistoric Class Reference

An Implementation of the IBusSolidHistoricTimeTable interface for the Reading Buses API.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Reading↩ Buses.RbTimeTableSolidHistoric:



Static Public Member Functions

- static operator RbTimeTableSolidHistoric (RbTimeTableHistoric inputTimeTable)

  Converts between a non-solid to solid object.

Properties

- DateTime **ActArrivalTime** [get]
- DateTime **ActDepartureTime** [get]

Additional Inherited Members

5.43.1 Detailed Description

An Implementation of the IBusSolidHistoricTimeTable interface for the Reading Buses API.

5.43.2 Member Function Documentation

5.43.2.1 operator RbTimeTableSolidHistoric()

```
static Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTime↩
TableSolidHistoric.operator RbTimeTableSolidHistoric (
            RbTimeTableHistoric inputTimeTable )  [explicit], [static]
```

Converts between a non-solid to solid object.

Parameters

inputTimeTable	The non-solid input record.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/ReadingBuses/RB_TimeTableSolidHistoric.cs

5.44 Timetable_Optimisation_Recommendations.Properties.Resources Class Reference

A strongly-typed resource class, for looking up localized strings, etc.

Properties

- static global::System.Resources.ResourceManager ResourceManager [get]

  Returns the cached ResourceManager instance used by this class.
- static global::System.Globalization.CultureInfo Culture [get, set]

  Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.

- static string AverageLateness [get]

  Looks up a localized string similar to Average Lateness: {0}.
- static string Cluster_NoOfDays [get]

  Looks up a localized string similar to Number of days associated : #.
- static string OnTimePercentage [get]

  Looks up a localized string similar to On Time Percentage: {0}.
- static string ServiceString [get]

  Looks up a localized string similar to Service {0}.

5.44.1 Detailed Description

A strongly-typed resource class, for looking up localized strings, etc.

5.44.2 Property Documentation

5.44.2.1 AverageLateness

```
string Timetable_Optimisation_Recommendations.Properties.Resources.AverageLateness [static],
[get]
```

Looks up a localized string similar to Average Lateness: {0}.

5.44.2.2 Cluster_NoOfDays

```
string Timetable_Optimisation_Recommendations.Properties.Resources.Cluster_NoOfDays [static],
[get]
```

Looks up a localized string similar to Number of days associated : #.

5.44.2.3 Culture

```
global.System.Globalization.CultureInfo Timetable_Optimisation_Recommendations.Properties.↩
Resources.Culture [static], [get], [set]
```

Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.

5.44.2.4 OnTimePercentage

```
string Timetable_Optimisation_Recommendations.Properties.Resources.OnTimePercentage [static],
[get]
```

Looks up a localized string similar to On Time Percentage: {0}.

5.44.2.5 ResourceManager

```
global.System.Resources.ResourceManager Timetable_Optimisation_Recommendations.Properties.↩
Resources.ResourceManager [static], [get]
```

Returns the cached ResourceManager instance used by this class.

5.44.2.6 ServiceString

```
string Timetable_Optimisation_Recommendations.Properties.Resources.ServiceString [static],
[get]
```

Looks up a localized string similar to Service {0}.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Properties/Resources.↩ Designer.cs

5.45 Timetable_Optimisation_Recommendations.Route_Analyser.↩ RouteSegment Class Reference

Route Segment is set of consecutive stops that two services share, the primary service, identified in the Route↩ SegmenetFinder and the secondary service that also shares it.

Public Member Functions

- int LengthOfSegment ()

  Finds the length of the segment of stops.

Properties

- IBusService SecondaryService [get]

  The other service that shares the route segment.
- List< IBusStop > Stops = new() [get]

  Defines the set of consecutive stops that makes up the route segment.

5.45.1 Detailed Description

Route Segment is set of consecutive stops that two services share, the primary service, identified in the Route↩ SegmenetFinder and the secondary service that also shares it.

5.45.2 Member Function Documentation

5.45.2.1 LengthOfSegment()

```
int Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment.LengthOfSegment ( )
```

Finds the length of the segment of stops.

Returns

The length of the segment.

5.45.3 Property Documentation

5.45.3.1 SecondaryService

```
IBusService Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment.Secondary↩
Service  [get]
```

The other service that shares the route segment.

5.45.3.2 Stops

```
List<IBusStop> Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment.Stops =
new()  [get]
```

Defines the set of consecutive stops that makes up the route segment.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Route Analyser/Route↩ Segment.cs

5.46 Timetable_Optimisation_Recommendations.Route_Analyser.↩ RouteSegmentCollection Class Reference

The RouteSegmentCollection class manages the results of the RouteSegmentFinder, and provides the logic for part of the GUI, which lets the user add or remove a service of interest.

Public Member Functions

- async Task InitialiseAsync (RouteSegmentFinder finderObj, IProgress< ProgressReporting >? progress=null)

  Used to actually initialise the object, given a route segment finder and a progress reporter.
- async Task AddService (IBusService service, IProgress< ProgressReporting >? progress)

  Adds in a specific service to be included as part of the search.
- void RemoveServiceAsync (IBusService service)

  Removes a service from the search.
- IBusService[ ] GetServices (IBusStop s1, IBusStop s2)

  Given to bus stops, stop 1 and stop 2 find all the services that go between it. There is no guarantee that they do so consecutively
- async Task< IBusService[ ]> GetAllServicesAsync (IProgress< ProgressReporting >? progress)

  Gets an array of all the bus services that are apart of all route segments.
- async Task< IBusStop[ ]> GetAllSharedBusStopsAsync (IProgress< ProgressReporting >? progress)

  Gets all stops that are apart of a shared route segment.
- IBusStop[ ] GetSharedBusStopsAsync ()

  Gets an array of bus stops apart of the shared route-segment, including only stops that contains services that has been included. Unlike the GetAllSharedBusStopsAsync method above.

Properties

- Dictionary< IBusStop, List< IBusService > > ServicesAtStopOfInterest = new() `[get]`

  value>Used by the GUI to store the included services.
- ObservableCollection< IBusService > IncludedServices = new() `[get]`

  value>Used by the GUI to store the excluded services.
- ObservableCollection< IBusService > ExcludedServices = new() `[get]`

  value>Contains an array of all the services across all of the route segments.

5.46.1 Detailed Description

The RouteSegmentCollection class manages the results of the RouteSegmentFinder, and provides the logic for part of the GUI, which lets the user add or remove a service of interest.

5.46.2 Member Function Documentation

5.46.2.1 AddService()

```
async Task Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection.Add↩
Service (
            IBusService service,
            IProgress< ProgressReporting >?  progress )
```

Adds in a specific service to be included as part of the search.

Parameters

service	The service that you wish to add to the search space
progress	The progress of the task, shouldn't ever take very long.

Returns

5.46.2.2 GetAllServicesAsync()

```
async Task<IBusService[]> Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment↩
Collection.GetAllServicesAsync (
            IProgress< ProgressReporting >?  progress )
```

Gets an array of all the bus services that are apart of all route segments.

Parameters

progress	The progress reporter for this task.

Returns

An array of all services in the route segment.

This should only be used to check if you have all services cached and if you want to add all services. This does NOT give all services that are actually included. (it also includes excluded)

5.46.2.3 GetAllSharedBusStopsAsync()

```
async Task<IBusStop[]> Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment↩
Collection.GetAllSharedBusStopsAsync (
            IProgress< ProgressReporting >?  progress )
```

Gets all stops that are apart of a shared route segment.

Parameters

progress	The progress reporter for the task.

Returns

An array of bus stops that are apart of a shared route segment.

5.46.2.4 GetServices()

```
IBusService [] Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection.↩
GetServices (
            IBusStop s1,
            IBusStop s2 )
```

Given to bus stops, stop 1 and stop 2 find all the services that go between it. There is no guarantee that they do so consecutively

Parameters

s1	Bus Stop 1
s2	Bus Stop 2

Returns

5.46.2.5 GetSharedBusStopsAsync()

```
IBusStop [] Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection.Get↩
SharedBusStopsAsync ( )
```

Gets an array of bus stops apart of the shared route-segment, including only stops that contains services that has been included. Unlike the GetAllSharedBusStopsAsync method above.

Returns

All stops that are apart of a shared route segment.

5.46.2.6 InitialiseAsync()

```
async Task Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection.↩
InitialiseAsync (
            RouteSegmentFinder finderObj,
            IProgress< ProgressReporting >?  progress = null )
```

Used to actually initialise the object, given a route segment finder and a progress reporter.

Parameters

finderObj	
progress	

Returns

5.46.2.7 RemoveServiceAsync()

```
void Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection.Remove↩
ServiceAsync (
            IBusService service )
```

Removes a service from the search.

Parameters

service	The service that you wish to remove.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Route Analyser/Route↩ SegmentCollection.cs

5.47 Timetable_Optimisation_Recommendations.Route_Analyser.↩ RouteSegmentFinder Class Reference

The route segment finder class takes in a Bus Service (known as the primary service) and a tolerance. It then finds any service which shares a common-route segment with it, to the specified minimum segment length tolerance.

Public Member Functions

- RouteSegmentFinder (IBusService primaryService, int? routeSegmentMinimum=null)

  Default constructor for the route segment finder.
- async Task< List< RouteSegment > > FindSharedRouteSegmentsAsync (IProgress< ProgressReporting >? progress)

  Finds any services which might have a shared route segment and what the segment contains.
- async Task< IBusService[]> GetServicesInSegments (IProgress< ProgressReporting >? progress)

  Finds all the distinct services which shared a route segment with us. One secondary service might have multiple segments, if they diverged and re-join or inbound and outbound.

Properties

- IBusService PrimaryService [get]

5.47.1 Detailed Description

The route segment finder class takes in a Bus Service (known as the primary service) and a tolerance. It then finds any service which shares a common-route segment with it, to the specified minimum segment length tolerance.

The primary purpose of this is to find the shared bus-corridors.

5.47.2 Constructor & Destructor Documentation

5.47.2.1 RouteSegmentFinder()

```
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder.RouteSegmentFinder (
            IBusService primaryService,
            int?  routeSegmentMinimum = null )
```

Default constructor for the route segment finder.

Parameters

primaryService	The service you want to find common route segments with
routeSegmentMinimum	the minimum length of a route segment to count, only used if you wish to override settings variable.

5.47.3 Member Function Documentation

5.47.3.1 FindSharedRouteSegmentsAsync()

```
async Task<List<RouteSegment> > Timetable_Optimisation_Recommendations.Route_Analyser.Route↩
SegmentFinder.FindSharedRouteSegmentsAsync (
            IProgress< ProgressReporting >?  progress )
```

Finds any services which might have a shared route segment and what the segment contains.

Returns

A list of found route segments.

5.47.3.2 GetServicesInSegments()

```
async Task<IBusService[]> Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment↩
Finder.GetServicesInSegments (
            IProgress< ProgressReporting >?  progress )
```

Finds all the distinct services which shared a route segment with us. One secondary service might have multiple segments, if they diverged and re-join or inbound and outbound.

Returns

An array of distinct services that have a route segment with the primary service.

This is includes it self, the primary service.

5.47.4 Property Documentation

5.47.4.1 PrimaryService

```
IBusService Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder.Primary↩
Service  [get]
```

The primary service for which you want to find services that share a route segment with.

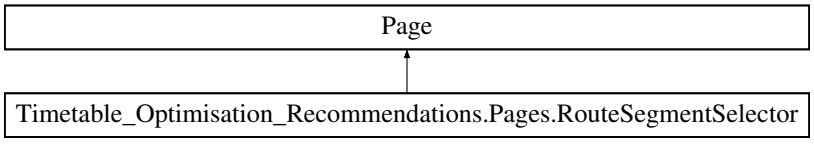
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Route Analyser/Route↩ SegmentFinder.cs

5.48 Timetable_Optimisation_Recommendations.Pages.RouteSegment↩ Selector Class Reference

The route-segment selector page finds other services that share a common route-segment with the primary service. And then allows the user to accept secondary services to also optimism their timetables for.

Inheritance diagram for Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector:



Public Member Functions

- RouteSegmentSelector (Cluster cluster)

  The default constructor, takes in the dates from the other services.

Properties

- RouteSegmentCollection Collection = new() [get]

  value>Used to update the GUI and report progress.
- ProgressReporting **Reporter** = new() [get]

5.48.1 Detailed Description

The route-segment selector page finds other services that share a common route-segment with the primary service. And then allows the user to accept secondary services to also optimism their timetables for.

5.48.2 Constructor & Destructor Documentation

5.48.2.1 RouteSegmentSelector()

```
Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector.RouteSegmentSelector (
            Cluster cluster )
```

The default constructor, takes in the dates from the other services.

Parameters

cluster	Dates of interest for the search.

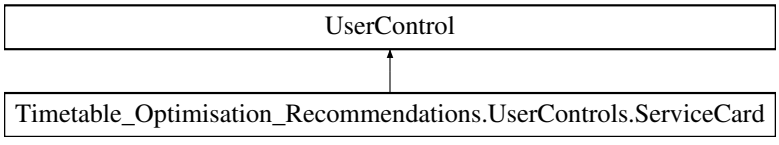
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Pages/RouteSegmentSelector.xaml.cs

5.49 Timetable_Optimisation_Recommendations.UserControls.ServiceCard Class Reference

The service card, use to display the single service.

Inheritance diagram for Timetable_Optimisation_Recommendations.UserControls.ServiceCard:



5.49.1 Detailed Description

The service card, use to display the single service.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/UserControls/ServiceCard.xaml.cs

5.50 Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator Class Reference

The service cohesion evaluator works with the Evaluator class to calculate how well a services timetable works, with another service that shares a common route segment. By assigning a blame value to each timetable record based on how un-cohesive it is.

Public Member Functions

- ServiceCohesionEvaluator (TimeTableEvaluator evaluator, double? dominance=null)

  The default constructor for the service cohesion evaluator object, Which takes in the parent evaluator and the fixed dominance value.
- void FindBlameServiceCohesion (Solution solution)

  Works out the service cohesion for services that share a common route-segments for specific bus stops.

Properties

- static double Dominance = Properties.Settings.Default.CohesionDominance `[get]`

  The fixed dominance value.

5.50.1 Detailed Description

The service cohesion evaluator works with the Evaluator class to calculate how well a services timetable works, with another service that shares a common route segment. By assigning a blame value to each timetable record based on how un-cohesive it is.

5.50.2 Constructor & Destructor Documentation

5.50.2.1 ServiceCohesionEvaluator()

```
Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator.ServiceCohesionEvaluator (
            TimeTableEvaluator evaluator,
            double?  dominance = null )
```

The default constructor for the service cohesion evaluator object, Which takes in the parent evaluator and the fixed dominance value.

Parameters

evaluator	Takes in the main evaluator object as reference
dominance	The dominance value for the cohesion blame value. Only use if you wish to override the settings value.

5.50.3 Member Function Documentation

5.50.3.1 FindBlameServiceCohesion()

```
void Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator.↩
FindBlameServiceCohesion (
            Solution solution )
```

Works out the service cohesion for services that share a common route-segments for specific bus stops.

Returns

A dictionary, where bus service is key, and the value is an array of tuples of timetable records and blame values. Where.....

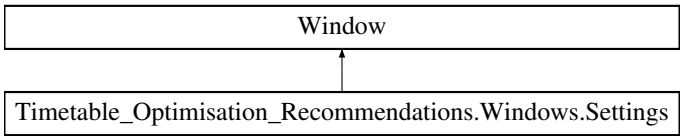
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Evaluator/Service↩ CohesionEvaluator.cs

5.51 Timetable_Optimisation_Recommendations.Windows.Settings Class Reference

The main settings page for the application.

Inheritance diagram for Timetable_Optimisation_Recommendations.Windows.Settings:



Public Member Functions

- Settings ()

  The default constructor, sets up the GUI.

5.51.1 Detailed Description

The main settings page for the application.

5.51.2 Constructor & Destructor Documentation

5.51.2.1 Settings()

```
Timetable_Optimisation_Recommendations.Windows.Settings.Settings ( )
```

The default constructor, sets up the GUI.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Windows/Settings.↩ xaml.cs

5.52 Timetable_Optimisation_Recommendations.Timetable_Evaluator.↩ SlackTimeEvaluator Class Reference

The slack time evaluator works with the Evaluator class to calculate if the timetable has excessive slack (Or not enough). By assigning a blame value to each timetable record based on how disruptive it is to the timetable.

Public Member Functions

- SlackTimeEvaluator (TimeTableEvaluator evaluator, double? dominance=null)

  Default constructor for the slack time evaluator.
- async Task FindBlameSlackTime (Solution solution, IProgress< AdvancedProgressReporting >? progress=null)

  Will calculate the slack time blame values for all of the services needed in the evaluator
- async Task FindSingleBlameSlackTime (BlamedBusTimeTable[ ] serviceTimetable, IProgress< ProgressReporting >? progress=null)

  Given a day worth of timetable data go through every record and then assign a blame value to each record. Return this list of tuples to then decide what problem areas to focus upon.

Static Public Member Functions

- static void StandardiseSolution (Solution unstandardised)

  Takes in an array of unstandardised blame records, and then standardises their slack time value, such that it can be compared to other blame values. It will also adjust this value, such that the pareto-dominance acts on the value.

Properties

- static double Dominance = Properties.Settings.Default.SlackTimeDominance `[get]`

  The fixed dominance value.

5.52.1 Detailed Description

The slack time evaluator works with the Evaluator class to calculate if the timetable has excessive slack (Or not enough). By assigning a blame value to each timetable record based on how disruptive it is to the timetable.

5.52.2 Constructor & Destructor Documentation

5.52.2.1 SlackTimeEvaluator()

```
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator.SlackTime↩
Evaluator (
            TimeTableEvaluator evaluator,
            double?  dominance = null )
```

Default constructor for the slack time evaluator.

Parameters

evaluator	The evaluator object, which stores the current proposed solution.
dominance	The dominance value for the slack time blame value. Only use if you wish to override the settings value.

5.52.3 Member Function Documentation

5.52.3.1 FindBlameSlackTime()

```
async Task Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator.↩
FindBlameSlackTime (
            Solution solution,
            IProgress< AdvancedProgressReporting >?  progress = null )
```

Will calculate the slack time blame values for all of the services needed in the evaluator

Parameters

solution	The solution you wish to apply slack time evaluator too. By defualt this will be the one in the evualtor.
progress	Used to report back the total progress of the task.

Returns

Once completed all blame records will have a slack value blame.

5.52.3.2 FindSingleBlameSlackTime()

```
async Task Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator.↩
FindSingleBlameSlackTime (
            BlamedBusTimeTable[] serviceTimetable,
            IProgress< ProgressReporting >?  progress = null )
```

Given a day worth of timetable data go through every record and then assign a blame value to each record. Return this list of tuples to then decide what problem areas to focus upon.

Parameters

serviceTimetable	A days worth of timetable data.
progress	A progress reporter to update GUI.

If you call this function you MUST standardised the results after wards. This isn't done within the function itself encase you call it several times over. Then you only need to call it after you last call to the function.

Returns

An array of timetable records and blame values.

5.52.3.3 StandardiseSolution()

```
static void Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator.↩
StandardiseSolution (
            Solution unstandardised )  [static]
```

Takes in an array of unstandardised blame records, and then standardises their slack time value, such that it can be compared to other blame values. It will also adjust this value, such that the pareto-dominance acts on the value.

Parameters

unstandardised	An array of unstandardised blame values.

Changing any timetable record, would effect all other blame records, regardless of if it was in the journey group or not. As such the unstandardised values needs to be cached.

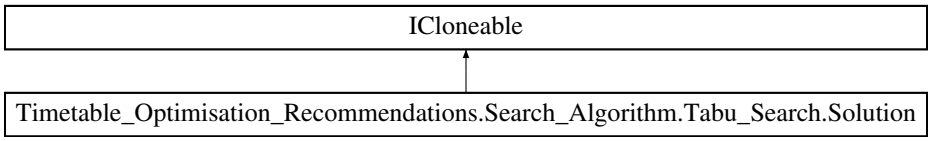
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Evaluator/Slack↩ TimeEvaluator.cs

5.53 Timetable_Optimisation_Recommendations.Search_Algorithm.↩ Tabu_Search.Solution Class Reference

Used to represent a single solution to the problem, the actual solution is stored in a dictionary, where the service is the key and an array of timetable records is the value.

Inheritance diagram for Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution:



Public Member Functions

- Solution (Dictionary< IBusService, BlamedBusTimeTable[]> busTimetables)

  Default constructor takes in a solution dictionary and stores it.
- double ObjectiveFunctionValue ()

  Returns back an objective function value for the current solution, a lower score is better. Values can only be compared against solutions to the same problem. These are not standardised between different searches.
- void CalculateTotalBlames ()

  Goes through every record in the solution space and updates their weights to the new total weight.
- Solution ReplaceMove (Move move)

  Creates a deep-clone of thew current solution and then replaces a move in the solution space. Returns the copy of the solution with the altered move.
- int ScoreOfService (IBusService service)

  Returns the objective function score of in single service in the solution. This can be used to work out how much one service has improved.
- object Clone ()

  Creates a deep-clone of the object, this will shallow-clone the object and then deep-clone the bus-timetable solution dictionary.

Properties

- Dictionary< IBusService, BlamedBusTimeTable[]> BusTimeTables `[get]`

  Stores the solution, as a set of timetable records for a day with an associated service.

5.53.1 Detailed Description

Used to represent a single solution to the problem, the actual solution is stored in a dictionary, where the service is the key and an array of timetable records is the value.

5.53.2 Constructor & Destructor Documentation

5.53.2.1 Solution()

```
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.Solution (
            Dictionary< IBusService, BlamedBusTimeTable[]> busTimetables )
```

Default constructor takes in a solution dictionary and stores it.

Parameters

busTimetables	

5.53.3 Member Function Documentation

5.53.3.1 CalculateTotalBlames()

```
void Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.Calculate↩
TotalBlames ( )
```

Goes through every record in the solution space and updates their weights to the new total weight.

5.53.3.2 Clone()

```
object Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.Clone ( )
```

Creates a deep-clone of the object, this will shallow-clone the object and then deep-clone the bus-timetable solution dictionary.

Returns

A deep-clone of "this" object.

5.53.3.3 ObjectiveFunctionValue()

```
double Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.Objective↩
FunctionValue ( )
```

Returns back an objective function value for the current solution, a lower score is better. Values can only be compared against solutions to the same problem. These are not standardised between different searches.

Returns

An object function value, lower the score the better the solution. Zero being the "perfect" timetable.

5.53.3.4 ReplaceMove()

```
Solution Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.Replace↩
Move (
            Move move )
```

Creates a deep-clone of thew current solution and then replaces a move in the solution space. Returns the copy of the solution with the altered move.

Parameters

move	The move to replace the service with.

Returns

5.53.3.5 ScoreOfService()

```
int Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.ScoreOf↩
Service (
            IBusService service )
```

Returns the objective function score of in single service in the solution. This can be used to work out how much one service has improved.

Parameters

service	The service in the solution to generate a score for. If not in the solution 0.

Returns

Objective score of one service in the solution.

5.53.4 Property Documentation

5.53.4.1 BusTimeTables

```
Dictionary<IBusService, BlamedBusTimeTable[]> Timetable_Optimisation_Recommendations.Search↩
_Algorithm.Tabu_Search.Solution.BusTimeTables  [get]
```

Stores the solution, as a set of timetable records for a day with an associated service.

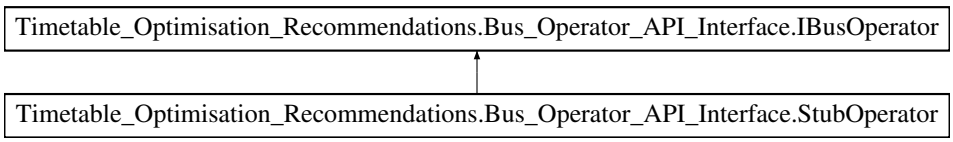
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Search Algorithm/↩ Tabu Search/Solution.cs

5.54 Timetable_Optimisation_Recommendations.Bus_Operator_API_↩ Interface.StubOperator Class Reference

The default operator if none is selected, this is mainly used to satisfy the null-ability requirement of C# then anything else.

Inheritance diagram for Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator:



Public Member Functions

- void ForceUpdateCache ()

  Forces the current data stored in the bus operator object to be saved into Cache. You would need to do this if you've made some lazy API requests.
- IBusStop GetLocation (string atcoCode)

  Get a bus stop location based upon a bus stops location code
- IBusService GetService (string serviceNumber)

  Returns a service which matches the Service Number passed.
- IBusService[ ] GetServices ()

  Gets an array of all the IBusServices Objects.
- void InvalidateCache ()

  Deletes any Cache data stored, use this only if you need to force new data cache.
- bool IsLocation (string actoCode)

  Checks to see if the atco code for the bus stop exists in the API feed or not.
- bool IsService (string serviceNumber)

  Checks to see if a service of that number exists or not in the API feed.

5.54.1 Detailed Description

The default operator if none is selected, this is mainly used to satisfy the null-ability requirement of C# then anything else.

5.54.2 Member Function Documentation

5.54.2.1 ForceUpdateCache()

```
void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.Force↩
UpdateCache ( )
```

Forces the current data stored in the bus operator object to be saved into Cache. You would need to do this if you've made some lazy API requests.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.54.2.2 GetLocation()

```
IBusStop Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.Get↩
Location (
            string atcoCode )
```

Get a bus stop location based upon a bus stops location code

Parameters

atcoCode	The code of the bus stop

Returns

A Bus Stop object for the Atco Code specified.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.54.2.3 GetService()

```
IBusService Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.↩
GetService (
            string serviceNumber )
```

Returns a service which matches the Service Number passed.

Parameters

serviceNumber	The service number/ID for the service you wish to be returned eg: 17 or 22.

Returns

The services matching the ID.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.54.2.4 GetServices()

```
IBusService [] Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.↩
GetServices ( )
```

Gets an array of all the IBusServices Objects.

Returns

An array of all the bus services.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.54.2.5 InvalidateCache()

```
void Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.Invalidate↩
Cache ( )
```

Deletes any Cache data stored, use this only if you need to force new data cache.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.54.2.6 IsLocation()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.IsLocation
(
            string atcoCode )
```

Checks to see if the atco code for the bus stop exists in the API feed or not.

Parameters

atcoCode	The ID Code for a bus stop.

Returns

True or False depending on if the stop is in the API feed or not.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

5.54.2.7 IsService()

```
bool Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator.IsService
(
            string serviceNumber )
```

Checks to see if a service of that number exists or not in the API feed.

Parameters

serviceNumber	The service number to find.

Returns

True or False for if a service is the API feed or not.

Implements Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Bus Operator API Interface/StubOperator.cs

5.55 Timetable_Optimisation_Recommendations.Search_Algorithm.↩ Tabu_Search.TabuList Class Reference

TabuList keeps track of what moves are tabu/invalid and what moves can now be performed. Moves are tabu while the tabu tenure is greater than zero.

Public Member Functions

- TabuList (int? tabuTenure=null)

  An optional constructor used to manually specify the TabuTenure, only use if you want to override the settings value.
- void SetTabu (Move move)

  Once a move has been accepted it needs to be made tabu, and previous tabu records updated.
- bool IsTabu (Move move)

  Returns true if the move is going to be editing a tabu-timetable record or not. Therefore, false is an allowed move.
- bool IsTabu (IBusTimeTable record)

  Returns true if the record is considered tabu or not.
- void FreeUpTabuListEarly ()

  Used if all of the search space has become tabu, this should only happen if the user has set a really long tabu tenure or if they have a really small search space. It will go through and decrement all keys by one in the hopes that some are removed.

5.55.1 Detailed Description

TabuList keeps track of what moves are tabu/invalid and what moves can now be performed. Moves are tabu while the tabu tenure is greater than zero.

5.55.2 Constructor & Destructor Documentation

5.55.2.1 TabuList()

```
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList.TabuList (
            int?  tabuTenure = null )
```

An optional constructor used to manually specify the TabuTenure, only use if you want to override the settings value.

Parameters

tabuTenure	

5.55.3 Member Function Documentation

5.55.3.1 FreeUpTabuListEarly()

```
void Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList.FreeUpTabu↩
ListEarly ( )
```

Used if all of the search space has become tabu, this should only happen if the user has set a really long tabu tenure or if they have a really small search space. It will go through and decrement all keys by one in the hopes that some are removed.

5.55.3.2 IsTabu() [1/2]

```
bool Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList.IsTabu (
            IBusTimeTable record )
```

Returns true if the record is considered tabu or not.

Parameters

record	A timetable record.

Returns

True if this record is tabu.

5.55.3.3 IsTabu() [2/2]

```
bool Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList.IsTabu (
            Move move )
```

Returns true if the move is going to be editing a tabu-timetable record or not. Therefore, false is an allowed move.

Parameters

move	The move to evaluate if it's tabu or not.

Returns

True or False for if it's tabu or not.

5.55.3.4 SetTabu()

```
void Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList.SetTabu (
            Move move )
```

Once a move has been accepted it needs to be made tabu, and previous tabu records updated.

Parameters

move	The new move just been performed.

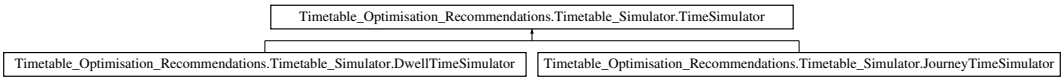
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Search Algorithm/↩ Tabu Search/TabuList.cs

5.56 Timetable_Optimisation_Recommendations.Timetable_Simulator.↩ TimeSimulator Class Reference

Shared common code used between both the Journey Time Simulator and the Dwell Time Simulator. Manila the weighted averaging code and accuracy measures.

Inheritance diagram for Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator:



Static Protected Member Functions

- static double CalculateInverseWeight (double value1)

  Calculates the inverse of a value, but if the value is less than one just return one.
- static double CalculateInverseWeight (double value1, double value2)

  Calculates the inverse of the smallest of the two values, but if the smallest value is less than one, just return one.
- static TimeSpan GenerateWeightedAverage (List<(TimeSpan duration, double accuracyWeight)> times)

  Given a list of estimated durations and their estimated accuracy/ confidence level generate the weighted average of all the values.

5.56.1 Detailed Description

Shared common code used between both the Journey Time Simulator and the Dwell Time Simulator. Manila the weighted averaging code and accuracy measures.

5.56.2 Member Function Documentation

5.56.2.1 CalculateInverseWeight() [1/2]

```
static double Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator.↩
CalculateInverseWeight (
            double value1 )  [static], [protected]
```

Calculates the inverse of a value, but if the value is less than one just return one.

Parameters

value1	Value to inverse

Returns

Inverses the value, unless less than one, then return one.

5.56.2.2 CalculateInverseWeight() [2/2]

```
static double Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator.↩
CalculateInverseWeight (
            double value1,
            double value2 )  [static], [protected]
```

Calculates the inverse of the smallest of the two values, but if the smallest value is less than one, just return one.

Parameters

value1	Value one
value2	Value two

Returns

The inverse of the smallest of the two values, unless less than one, then return one.

5.56.2.3 GenerateWeightedAverage()

```
static TimeSpan Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator.↩
GenerateWeightedAverage (
            List<(TimeSpan duration, double accuracyWeight)> times )  [static], [protected]
```

Given a list of estimated durations and their estimated accuracy/ confidence level generate the weighted average of all the values.

Parameters

times	A list of tuples of estimated durations and accuracy

Returns

The new single weighted average of all the values.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Simulator/Time↩ Simulator.cs

5.57 Timetable_Optimisation_Recommendations.Timetable_Evaluator.↩ TimeTableEvaluator Class Reference

The main evaluator entry point, give it a set of dates where you want to use data from, an initial starting solution and a route-segment collection and then it can improve upon the timetable.

Public Member Functions

- TimeTableEvaluator (DateTime[] relatedDates, RouteSegmentCollection segmentCollection, Dictionary< IBusService, IBusTimeTable[]> busTimeTables)

  The default constructor for the main evaluator.
- async Task< bool > PerformIterationAsync (IProgress< AdvancedProgressReporting >? progress=null)

  Performs one iteration of the search algorithm, once it's completed the solution will have changed by one move.

Public Attributes

- Solution? **Solution**
- string

  Used to tell what the last solution score was and to show if it has been improving or not.

Properties

- DateTime[] RelatedDates `[get]`

  Dates when the timetables where the same in the year.
- Solution CurrentSolution `[get]`

  value>The initial start solution.
- Solution? StartSolution `[get]`

  value>The best solution found in any iteration
- List<(Move, int)> MovesMade = new() `[get]`

  value>Stores information about shared route-segment collections.
- RouteSegmentCollection **Collection** `[get]`

5.57.1 Detailed Description

The main evaluator entry point, give it a set of dates where you want to use data from, an initial starting solution and a route-segment collection and then it can improve upon the timetable.

5.57.2 Constructor & Destructor Documentation

5.57.2.1 TimeTableEvaluator()

```
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator.TimeTable↩
Evaluator (
            DateTime[] relatedDates,
            RouteSegmentCollection segmentCollection,
            Dictionary< IBusService, IBusTimeTable[]> busTimeTables )
```

The default constructor for the main evaluator.

Parameters

relatedDates	A set of dates where the timetables were the same and to look for data.
segmentCollection	A route-segment collection, to find the common-shared path segments.
busTimeTables	The current timetable/ initial proposed solution.

5.57.3 Member Function Documentation

5.57.3.1 PerformIterationAsync()

```
async Task<bool> Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator.↩
PerformIterationAsync (
            IProgress< AdvancedProgressReporting >?  progress = null )
```

Performs one iteration of the search algorithm, once it's completed the solution will have changed by one move.

Returns

Updates the solution by one move improving the solution. Returns true or false for if you can perform another subsequent move.

5.57.4 Member Data Documentation

5.57.4.1 string

```
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator.string
```

Used to tell what the last solution score was and to show if it has been improving or not.

Returns

A string to say the score/progress of the algorithm, along with a boolean value. True - Has improved since last move. False- Gotten worse since last move.

5.57.5 Property Documentation

5.57.5.1 RelatedDates

```
DateTime [] Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator.↩
RelatedDates [get]
```

Dates when the timetables where the same in the year.

value>The current proposed solution.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Evaluator/Time↩ TableEvaluator.cs

5.58 Timetable_Optimisation_Recommendations.Timetable_Analyser.↩ TimeTableGrouper Class Reference

The grouper class can be used to help find patterns in the timetable where several days shared the same timetable.

Public Member Functions

- TimeTableGrouper (IBusService service)

  The default constructor for the grouper.
- async Task< Cluster[ ]> FindGroupings (IProgress< ProgressReporting > progress, DateTime startDate, DateTime endDate)

  Finds an array of clusters, within the timetable data between two dates. If every day had a new timetable then you would have one cluster per day.

5.58.1 Detailed Description

The grouper class can be used to help find patterns in the timetable where several days shared the same timetable.

5.58.2 Constructor & Destructor Documentation

5.58.2.1 TimeTableGrouper()

```
Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTableGrouper.TimeTableGrouper (
            IBusService service )
```

The default constructor for the grouper.

Parameters

service	The service for which you wish to find groupings.

5.58.3 Member Function Documentation

5.58.3.1 FindGroupings()

```
async Task<Cluster[]> Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTable↩
Grouper.FindGroupings (
            IProgress< ProgressReporting > progress,
            DateTime startDate,
            DateTime endDate )
```

Finds an array of clusters, within the timetable data between two dates. If every day had a new timetable then you would have one cluster per day.

Parameters

progress	The progress for how far along it is.
startDate	The start date to find a group in.
endDate	The end date to find a group in.

Returns

An array of found groups.

This does currently NOT work for a service which operates over night and might have a different start and end day. This is assuming that the start and end day is on the same day.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Analyser/Time↩ TableGrouper.cs

5.59 Timetable_Optimisation_Recommendations.Request_Manager.↩ TimetableRetrieval Class Reference

Used to get a batch/ multiple-days worth of timetable data in a single query. However, this still has to be done sequentially because the Reading Buses API doesn't like having more than one database connection open at once.

Static Public Member Functions

- static async Task< IBusHistoricTimeTable[][]> GetHistoricTimeTableBatch (DateTime start, DateTime end, IBusService service, IProgress< double >? progress=null)

  Gets multiple days worth of historical time table data for a particular service between two date and times.
- static async Task< IBusTimeTable[][]> GetTimeTableBatch (DateTime start, DateTime end, IBusService service, IProgress< ProgressReporting >? progress=null)

  Gets multiple days worth of planned time table data for a particular service between two dates and times.
- static async Task< IBusSolidHistoricTimeTable[][]> GetHistoricTimeTableBatch (DateTime[] cluster, IBusStop stop, IProgress< double >? progress=null)

  Gets multiple days worth of historical time table data for a particular service between two date and times.
- static async Task< IBusSolidHistoricTimeTable[][]> **GetHistoricTimeTableBatch** (DateTime[] cluster, IBusService service, IProgress< double >? progress=null)

5.59.1 Detailed Description

Used to get a batch/ multiple-days worth of timetable data in a single query. However, this still has to be done sequentially because the Reading Buses API doesn't like having more than one database connection open at once.

5.59.2 Member Function Documentation

5.59.2.1 GetHistoricTimeTableBatch() [1/2]

```
static async Task<IBusHistoricTimeTable[][]> Timetable_Optimisation_Recommendations.Request↩
_Manager.TimetableRetrieval.GetHistoricTimeTableBatch (
            DateTime start,
            DateTime end,
            IBusService service,
            IProgress< double >?  progress = null )  [static]
```

Gets multiple days worth of historical time table data for a particular service between two date and times.

Parameters

progress	Used to return back to the GUI the current progress of the task.
start	The start date, should be oldest into the past.
end	The end date for when you want to go up to, inclusive.
service	The service for which you want historical time table data for.

Returns

Gets all the days of historic timetable data between the two dates inclusive.

The days will not necessarily be in order, they are in the order in which the API sent data. If the ordering is important you must re-order it.

5.59.2.2 GetHistoricTimeTableBatch() [2/2]

```
static async Task<IBusSolidHistoricTimeTable[][]> Timetable_Optimisation_Recommendations.↩
Request_Manager.TimetableRetrieval.GetHistoricTimeTableBatch (
            DateTime[] cluster,
            IBusStop stop,
            IProgress< double >?  progress = null )  [static]
```

Gets multiple days worth of historical time table data for a particular service between two date and times.

Parameters

progress	Used to return back to the GUI the current progress of the task.
stop	The stop for which you want historical time table data for.
cluster	Used to get back data for a service at a specif cluster of dates.

Returns

5.59.2.3 GetTimeTableBatch()

```
static async Task<IBusTimeTable[][]> Timetable_Optimisation_Recommendations.Request_Manager.↩
TimetableRetrieval.GetTimeTableBatch (
            DateTime start,
            DateTime end,
            IBusService service,
            IProgress< ProgressReporting >?  progress = null )  [static]
```

Gets multiple days worth of planned time table data for a particular service between two dates and times.

Parameters

progress	Used to return back to the GUI the current progress of the task.
start	The start date, should be oldest into the past.
end	The end date for when you want to go up to, inclusive.
service	The service for which you want planned time table data for.

Returns

Gets all the days of timetable data between the two dates inclusive.

The days will not necessarily be in order, they are in the order in which the API sent data. If the ordering is important you must re-order it.

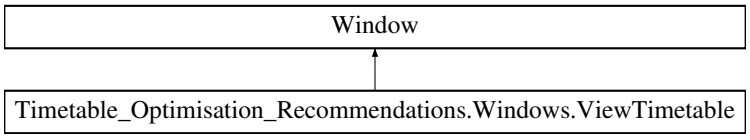
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Request Manager/Timetable Retrieval.cs

5.60 Timetable_Optimisation_Recommendations.Windows.View↩ Timetable Class Reference

Used to display the timetable to the user.

Inheritance diagram for Timetable_Optimisation_Recommendations.Windows.ViewTimetable:



Public Member Functions

- ViewTimetable (IBusTimeTable[ ] records, IBusService service)

  The default constructor for the timetable viewer.

5.60.1 Detailed Description

Used to display the timetable to the user.

5.60.2 Constructor & Destructor Documentation

5.60.2.1 ViewTimetable()

```
Timetable_Optimisation_Recommendations.Windows.ViewTimetable.ViewTimetable (
            IBusTimeTable[] records,
            IBusService service )
```

The default constructor for the timetable viewer.

Parameters

records	The timetable to display.
service	The service it is about.

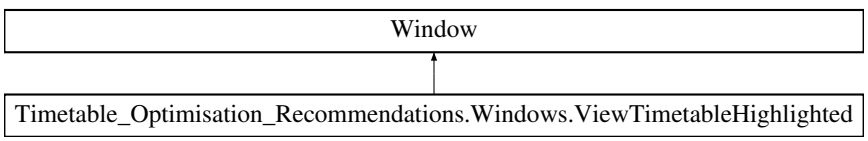
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Windows/ViewTimetable.xaml.cs

5.61 Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted Class Reference

Interaction logic for ViewTimetable.xaml

Inheritance diagram for Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted:



Public Member Functions

- ViewTimetableHighlighted (BlamedBusTimeTable[ ] records, IBusService service, string percentageChange, Move[ ]? moves=null)

  The default constructor for the view timetable highlights window.

5.61.1 Detailed Description

Interaction logic for ViewTimetable.xaml

5.61.2 Constructor & Destructor Documentation

5.61.2.1 ViewTimetableHighlighted()

```
Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted.ViewTimetableHighlighted
(
            BlamedBusTimeTable[] records,
            IBusService service,
            string percentageChange,
            Move?[] moves = null )
```

The default constructor for the view timetable highlights window.

Parameters

records	The timetable with blame values to display and highlight.
service	The service it pertains too.
percentageChange	The percentage improvement compared to the original timetable.
moves	The moves that the search algorithm made.

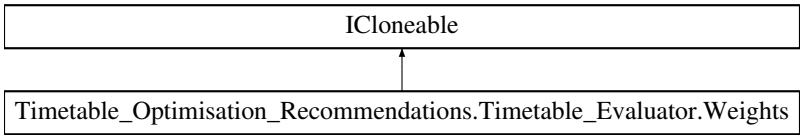
The documentation for this class was generated from the following file:

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Windows/ViewTimetableHighlighted.xaml.cs

5.62 Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights Class Reference

The weights class is used to store a target arrival and departure time, along with a raw and standardised weight for how much it should pull towards it.

Inheritance diagram for Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights:



Public Member Functions

- object Clone ()

  Performs a shallow copy of the weights object. But deep-copy would do the same as there are no object references in this class.

Properties

- double? Weight = null `[get, set]`

  The overall weight. This might be standardised, but will always be an absolute value.
- double? RawWeight `[get, set]`

  value>The Arrival Time value the weight is pulling towards.
- DateTime TargetSchArrivalTime `[get, set]`

  value>The Departure Time value the weight is pulling towards.
- DateTime **TargetSchDepartureTime** `[get, set]`

5.62.1 Detailed Description

The weights class is used to store a target arrival and departure time, along with a raw and standardised weight for how much it should pull towards it.

5.62.2 Member Function Documentation

5.62.2.1 Clone()

```
object Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights.Clone ( )
```

Performs a shallow copy of the weights object. But deep-copy would do the same as there are no object references in this class.

Returns

A copy of the object.

5.62.3 Property Documentation

5.62.3.1 Weight

```
double?  Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights.Weight = null
[get], [set]
```

The overall weight. This might be standardised, but will always be an absolute value.

value> The raw non-standardised and non-absolute weight used between iterations to regenerate the new standardised weight if possible. This is used mainly for speed and efficiency purposes rather than anything else.

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

- Timetable Optimisation Recommendations/Timetable Optimisation Recommendations/Timetable Evaluator/Weights.↩ cs

Index

AddDate
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster, 31
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.DateSpan, 37
AddRecords
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.PerformanceEvaluator, 77
AddService
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection, 104
AdvancedProgressReporting
Timetable_Optimisation_Recommendations.AdvancedProgressReporting, 18
AssociatedTimes
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster, 32
AverageLateness
Timetable_Optimisation_Recommendations.Properties.Resources, 101
BestMoveSelectorAsync
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.MoveSelector, 69
BlamedBusTimeTable
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, 21
BusTimeTables
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution, 118
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster, 32
CalculateGrouping
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster, 32
CalculateInverseWeight
Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator, 124, 126
CalculateTotalBlames
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution, 117
Clear
Timetable_Optimisation_Recommendations.AdvancedProgressReporting, 19
Timetable_Optimisation_Recommendations.ProgressReporting, 83
Clone
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution, 117
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Bla
Timetable_Optimisation_Recommendations.Timetable_Evaluator.We
Cluster
Timetable_Optimisation_Recommendations.Timetable_Analyser.Clus
Cluster_NoOfDays
Timetable_Optimisation_Recommendations.Properties.Resources, 104
ClusterCard
Timetable_Optimisation_Recommendations.UserControls.ClusterCar
ClusterId
Timetable_Optimisation_Recommendations.Timetable_Analyser.Clus
CouldBeSolid
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter
Culture
Timetable_Optimisation_Recommendations.Properties.Resources, 101
DateSelector
Timetable_Optimisation_Recommendations.Pages.DateSelector, 35
DateSpan
Timetable_Optimisation_Recommendations.Timetable_Analyser.Gro
Direction
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter
DownloadAllFilesNeeded
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Pre
DwellTimeSimulator
Timetable_Optimisation_Recommendations.Timetable_Simulator.Dw
End
Timetable_Optimisation_Recommendations.Timetable_Analyser.Gro
ErrorMessage
Timetable_Optimisation_Recommendations.UserControls.NotifyTask
TResult >, 74
EvaluateTimeTable
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Pre

Evaluator
Timetable_Optimisation_Recommendations.Pages.Evaluator, 41
Exception
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 74

FindBlameServiceCohesion
Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator, 112
FindBlameSlackTime
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator, 114
FindGroupings
Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTableGrouper, 130
FindSharedRouteSegmentsAsync
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder, 108
FindSingleBlameSlackTime
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator, 114
ForceUpdateCache
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator, 47
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator, 86
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, 119
FreeUpTabuListEarly
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList, 123

GenerateLatenessReport
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.PerformanceEvaluator, 77
GenerateNegibourhood
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.NeighbourhoodSolution, 72
GenerateWeightedAverage
Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator, 126
GetAllServicesAsync
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection, 105
GetAllSharedBusStopsAsync
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection, 105
GetArchivedTimeTable
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator, 50
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 55
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService, 90
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop, 93
GetAssociatedService
Timetable_Optimisation_Recommendations.Timetable_Analyser.Clus 32
GetHistoricTimeTableBatch
Timetable_Optimisation_Recommendations.Request_Manager.Timet 131
GetId
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 28
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 57
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 96
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Bla 22
GetLocation
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 47
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 86
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 119
GetLocations
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 50
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 90
GetService
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 47
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 88
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 120
GetServices
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 48
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 55
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 87
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 94
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 120
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSe 105
GetServicesInSegments
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSe 108
GetSharedBusStopsAsync
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSe 106
GetSolid
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 45
Timetable_Optimisation_Recommendations.Bus_Operator_API_Inter 99
GetStringAsync

Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 42
GetTimeTable
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 51
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService, 91
GetTimeTableBatch
Timetable_Optimisation_Recommendations.Request_Manager.TimetableRetrieval, 132
GetWeakArchivedTimeTable
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 55
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop, 94
Group
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 42
Grouping
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 43
GroupingAssociated
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 33
GroupingsOfClusters
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 43

Initialise
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator, 87
InitialiseAsync
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection, 106
InnerException
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 74
Instance
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperatorFactory, 26
InvalidateCache
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperator, 48
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator, 88
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.StubOperator, 120
IsArchivedTimeTableCached
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 51
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 56
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService, 91
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop, 94
IsCanceled
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 74
IsCancelledGroup,
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 75
IsFaulted
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 75
IsLate
Timetable_Optimisation_Recommendations.Timetable_Performance_Calculator.Group, 49
IsLocation
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 58
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop, 121
IsNotCompleted
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 75
IsOnTime
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator, 123
IsTimeTableCached
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 52
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 59
IsTimingPoint
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 52
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop, 92
IsWeekDay
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 37
JourneyCode
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 59
JourneyTime
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator, 60
JourneyTimeSimulator
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator, 62

Lateness
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord, 64
LatenessRecord
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord, 63
LengthOfSegment
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment, 103
Location
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, 24

MainPage
Timetable_Optimisation_Recommendations.Pages.MainPage, 66
MatchDirection
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub, 28
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable, 57
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTable, 96
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable, 22
MoveSelector
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.MoveSelector, 69
MovesMade
Timetable_Optimisation_Recommendations.Pages.MovesMade, 70

NeighbourhoodSolution
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.NeighbourhoodSolution, 71
NotifyTaskCompletion
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 74

ObjectiveFunctionValue
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution, 117
OnPropertyChanged
Timetable_Optimisation_Recommendations.ProgressReporting, 83
OnTimePercentage
Timetable_Optimisation_Recommendations.Properties.Resources, 101
Operator
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusOperatorFactory, 26
operator RbTimeTableSolidHistoric
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableSolidHistoric, 100
Operators
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface, 10

PerformIterationAsync
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimetableEvaluator.LatenessRecord, 126
PreEvaluatorChecks
Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvaluatorChecks, 78
PreliminaryDataDownload
Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload, 80
PreviousPerformance
Timetable_Optimisation_Recommendations.Pages.PreviousPerformance, 81
PrimaryService
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment.PrimaryService, 109
ProduceEstimatedDwell
Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeSimulator.ProduceEstimatedDwell, 40
ProduceEstimatedTravelTimes
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeSimulator.ProduceEstimatedTravelTimes, 62
ProgressReporting
Timetable_Optimisation_Recommendations.ProgressReporting, 83
PropertyChanged
Timetable_Optimisation_Recommendations.ProgressReporting, 84
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 76
ProposedSchArrivalTime
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.ProposedSchArrivalTime, 22
ProposedSchDepartureTime
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable.ProposedSchDepartureTime, 23

RelatedDates
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimetableEvaluator.RelatedDates, 128
RemoveServiceAsync
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment.RemoveServiceAsync, 107
ReplaceMove
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution.ReplaceMove, 117
Reporter
Timetable_Optimisation_Recommendations.Pages.Evaluator.Reporter, 41
ResourceManager
Timetable_Optimisation_Recommendations.Properties.Resources.ResourceManager, 102
Result
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >.Result, 75
RouteSegmentFinder
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment.RouteSegmentFinder, 108
RouteSegmentSelector
Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector, 110

RunningBoard
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable, 59
ScoreOfService
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution, 118
SecondaryService
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment, 103
Service
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable, 59
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Move, 68
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessReport, 65
ServiceCardCollection
Timetable_Optimisation_Recommendations.Pages.PreviousPerformance, 81
ServiceCohesionEvaluator
Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceCohesionEvaluator, 111
ServiceId
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 53
ServiceString
Timetable_Optimisation_Recommendations.Properties.Resources, 102
SetOperatorAsync
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperatorFactory, 26
SetSuggestedToReal
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Operators, 23
SetTabu
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList, 123
Settings
Timetable_Optimisation_Recommendations.WindowsSettings, 113
SlackTimeEvaluator
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator, 114
SlackWeights
Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackWeights, 25
Solution
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.Solution, 116
StandardiseSolution
Timetable_Optimisation_Recommendations.Timetable_Optimiser.StandardiseSolution, 115
Start
Timetable_Optimisation_Recommendations.Timetable_Analyser.Location, 38
StartDate
Timetable_Optimisation_Recommendations.Pages.DateSelection, 35
Status
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 75
Stops
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegment, 103
string
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimetableEvaluator, 128
SubValue
Timetable_Optimisation_Recommendations.AdvancedProgressReporting, 19
Tabu_Search.Move, 68
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Search.TabuList, 122
Task
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 76
TaskCompletion
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 76
Timetable_Optimisation_Recommendations, 9
Timetable_Optimisation_Recommendations.AdvancedProgressReporting, 17
AdvancedProgressReporting, 18
Clear, 19
SubValue, 19
Update, 19
Timetable_Optimisation_Recommendations.App, 19
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface, 9
Direction, 10
OperatedBusTimeTable, 10
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusOperatorFactory, 25
IBusService, 26
Operator, 26
SetOperatorAsync, 26
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 27
GetId, 28
MatchDirection, 28
WeakIsStopSame, 28, 29
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.OperatedBusTimeTable, 29
WriteToCache, 30
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Operator, 45
CouldBeSolid, 45
GetSolid, 45
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.Operator, 46
ForceUpdateCache, 47
GetLocation, 47
GetService, 47
GetServices, 48
InvalidateCache, 48
IsLocation, 48

IsService, 49
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusService, 49
GetArchivedTimeTable, 50
GetLocations, 50
GetTimeTable, 51
IsArchivedTimeTableCached, 51
IsTimeTableCached, 52
IsWeakServiceSame, 52
ServiceId, 53
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusStop, 53
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable, 54
GetArchivedTimeTable, 55
GetServices, 55
GetWeakArchivedTimeTable, 55
IsArchivedTimeTableCached, 56
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IMainPage, 56
GetId, 57
IsTimingPoint, 59
JourneyCode, 59
MatchDirection, 57
RunningBoard, 59
Service, 59
WeakIsStopSame, 58
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IReadingBuses, 11
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusOperator, 85
ForceUpdateCache, 86
GetLocation, 86
GetService, 86
GetServices, 87
Initialise, 87
InvalidateCache, 88
IsLocation, 88
IsService, 88
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusService, 89
GetArchivedTimeTable, 90
GetLocations, 90
GetTimeTable, 91
IsArchivedTimeTableCached, 91
IsTimeTableCached, 91
IsWeakServiceSame, 92
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbBusStop, 92
GetArchivedTimeTable, 93
GetServices, 94
GetWeakArchivedTimeTable, 94
IsArchivedTimeTableCached, 94
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTable, 95
GetId, 96
MatchDirection, 96
WeakIsStopSame, 97
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableSolid, 98
CouldBeSolid, 98
GetSolid, 99
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableSolidHistoric, 99
operator RbTimeTableSolidHistoric, 100
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTableSolidHistoricTimeTable, 119
ForceUpdateCache, 119
GetService, 120
GetServices, 120
InvalidateCache, 120
IsLocation, 121
IsService, 121
Timetable_Optimisation_Recommendations.Pages, 11
Timetable_Optimisation_Recommendations.Pages.DateSelector, 34
DateSelector, 35
StartDate, 35
Timetable_Optimisation_Recommendations.Pages.Evaluator, 40
Evaluator, 41
Reporter, 41
Timetable_Optimisation_Recommendations.Pages.MainPage, 66
MainPage, 66
Timetable_Optimisation_Recommendations.Pages.MovesMade, 70
MovesMade, 70
Timetable_Optimisation_Recommendations.Pages.PreliminaryDataDownload, 79
PreliminaryDataDownload, 80
Timetable_Optimisation_Recommendations.Pages.PreviousPerformance, 80
PreviousPerformance, 81
ServiceCardCollection, 81
Timetable_Optimisation_Recommendations.Pages.RouteSegmentSelector, 109
RouteSegmentSelector, 110
Timetable_Optimisation_Recommendations.ProgressReporting, 82
Clear, 83
OnPropertyChanged, 83
ProgressReporting, 83
PropertyChanged, 84
Timetable_Optimisation_Recommendations.Properties, 12
Timetable_Optimisation_Recommendations.Properties.Resources, 100
AverageLateness, 101
Culture, 101
OnTimePercentage, 101
ResourceManager, 102
ServiceString, 102

Timetable_Optimisation_Recommendations.Request_Manager, 12
Timetable_Optimisation_Recommendations.Request_Manager.TimeTableRetrieval, 130
GetHistoricTimeTableBatch, 131
GetTimeTableBatch, 132
Timetable_Optimisation_Recommendations.Route_Analyser, 12
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection, 102
LengthOfSegment, 103
SecondaryService, 103
Stops, 103
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentCollection, 104
AddService, 104
GetAllServicesAsync, 105
GetAllSharedBusStopsAsync, 105
GetServices, 105
GetSharedBusStopsAsync, 106
InitialiseAsync, 106
RemoveServiceAsync, 107
Timetable_Optimisation_Recommendations.Route_Analyser.RouteSegmentFinder, 107
FindSharedRouteSegmentsAsync, 108
GetServicesInSegments, 108
PrimaryService, 109
RouteSegmentFinder, 108
Timetable_Optimisation_Recommendations.Search_Algorithm, 12
Timetable_Optimisation_Recommendations.Search_Algorithm.TabuSearch, 12
Timetable_Optimisation_Recommendations.Search_Algorithm.TabuSearchMove, 67
Service, 68
ToString, 68
Timetable_Optimisation_Recommendations.Search_Algorithm.TabuSearchTimeTable, 68
BestMoveSelectorAsync, 69
MoveSelector, 69
Timetable_Optimisation_Recommendations.Search_Algorithm.TabuSearchNeighbourhoodSolution, 71
GenerateNegibourhood, 72
NeighbourhoodSolution, 71
Timetable_Optimisation_Recommendations.Search_Algorithm.TabuSearchSolution, 115
BusTimeTables, 118
CalculateTotalBlames, 117
Clone, 117
ObjectiveFunctionValue, 117
ReplaceMove, 117
ScoreOfService, 118
Solution, 116
Timetable_Optimisation_Recommendations.Search_Algorithm.TabuSearchTabuList, 122
FreeUpTabuListEarly, 123
IsTabu, 123
SetTabu, 123
TabuList, 122
Timetable_Optimisation_Recommendations.Timetable_Analyser, 13
Timetable_Optimisation_Recommendations.Timetable_Analyser.Cluster, 30
AddDate, 31
AssociatedTimes, 32
BusTimeTables, 32
CalculateGrouping, 32
Cluster, 31
ClusterId, 32
GetAssociatedService, 32
GroupingAssociated, 33
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group, 41
GetStringAsync, 42
Group, 42
Grouping, 43
GroupingsOfClusters, 43
Timetable_Optimisation_Recommendations.Timetable_Analyser.Group.Da 35
AddDate, 37
DateSpan, 36
End, 38
IsWeekDay, 37
Start, 38
ToString, 37
TotalSpan, 38
Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTabl 129
FindGroupings, 130
TimeTableGrouper, 129
Timetable_Optimisation_Recommendations.Timetable_Evaluator, 13
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedE 20
BlamedBusTimeTable, 21
Clone, 21
GetId, 22
Location, 24
MatchDirectionNeighbourhoodSolution, 22
ProposedSchArrivalTime, 22
ProposedSchDepartureTime, 23
SetSuggestedToReal, 23
SlackWeights, 25
UpdateTimes, 23
UpdateTotalWeight, 23
WeakIsStopSame, 24
Timetable_Optimisation_Recommendations.Timetable_Evaluator.PreEvalu 78
DownloadAllFilesNeeded, 79
EvaluateTimeTable, 79
PreEvaluatorChecks, 78
Timetable_Optimisation_Recommendations.Timetable_Evaluator.ServiceC 111
FindBlameServiceCohesion, 112
ServiceCohesionEvaluator, 111

Timetable_Optimisation_Recommendations.Timetable_Evaluator.SlackTimeEvaluator, 113
FindBlameSlackTime, 114
FindSingleBlameSlackTime, 114
SlackTimeEvaluator, 114
StandardiseSolution, 115
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator, 127
PerformIterationAsync, 128
RelatedDates, 128
string, 128
TimeTableEvaluator, 127
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights, 134
Clone, 135
Weight, 135
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator, 14
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessRecord, 63
IsLate, 64
Lateness, 64
LatenessRecord, 63
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.LatenessReport, 64
Service, 65
Timetable_Optimisation_Recommendations.Timetable_Performance_Evaluator.PerformanceEvaluator, 76
AddRecords, 77
GenerateLatenessReport, 77
Timetable_Optimisation_Recommendations.Timetable_Simulator, 14
Timetable_Optimisation_Recommendations.Timetable_Simulator.DwellTimeOptimisation, 39
DwellTimeSimulator, 39
ProduceEstimatedDwell, 40
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime, 60
JourneyTime, 60
TravelTime, 61
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTimeOptimisation, 61
JourneyTimeSimulator, 62
ProduceEstimatedTravelTimes, 62
Timetable_Optimisation_Recommendations.Timetable_Simulator.TimeSimulator, 124
CalculateInverseWeight, 124, 126
GenerateWeightedAverage, 126
Timetable_Optimisation_Recommendations.UserControls, 14
Timetable_Optimisation_Recommendations.UserControls.ClusterCard, 33
ClusterCard, 33
Timetable_Optimisation_Recommendations.UserControls.LatenessSummary, 65
Timetable_Optimisation_Recommendations.UserControls.NotifyTaskCompletion< TResult >, 72
ErrorMessage, 74
ExceptionMessage, 74
InnerException, 74
IsCanceled, 74
IsCompleted, 75
IsFaulted, 75
IsNotCompleted, 75
IsSuccessfullyCompleted, 75
NotifyTaskCompletion, 74
PropertyChanged, 76
Result, 75
Status, 75
Task, 76
TaskCompletion, 76
Timetable_Optimisation_Recommendations.UserControls.ServiceCard, 110
Timetable_Optimisation_Recommendations.Windows, 15
Timetable_Optimisation_Recommendations.Windows.Highlight, 44
X, 44
Timetable_Optimisation_Recommendations.Windows.MainWindow, 67
Timetable_Optimisation_Recommendations.Windows.Settings, 112
Settings, 113
Timetable_Optimisation_Recommendations.Windows.ViewTimetable, 132
ViewTimetable, 133
Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted, 133
ViewTimetableHighlighted, 134
TimeTableEvaluator
Timetable_Optimisation_Recommendations.Timetable_Evaluator.TimeTableEvaluator, 127
TimeTableGrouper
Timetable_Optimisation_Recommendations.Timetable_Analyser.TimeTableGrouper, 129
ToString
Timetable_Optimisation_Recommendations.Search_Algorithm.Tabu_Node, 68
Timetable_Optimisation_Recommendations.Timetable_Analyser.Grouping, 37
TotalSpan
Timetable_Optimisation_Recommendations.Timetable_Analyser.Grouping, 38
TravelTime
Timetable_Optimisation_Recommendations.Timetable_Simulator.JourneyTime, 61
Update
Timetable_Optimisation_Recommendations.AdvancedProgressReporting, 19
Timetable_Optimisation_Recommendations.ProgressReporting, 84
UpdateTimes
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlameSlackTime, 23
UpdateTotalWeight

Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable,
23

ViewTimetable
Timetable_Optimisation_Recommendations.Windows.ViewTimetable,
133
ViewTimetableHighlighted
Timetable_Optimisation_Recommendations.Windows.ViewTimetableHighlighted,
134

WeakIsStopSame
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.BusTimeTableStub,
28, 29
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.IBusTimeTable,
58
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.ReadingBuses.RbTimeTable,
97
Timetable_Optimisation_Recommendations.Timetable_Evaluator.BlamedBusTimeTable,
24
Weight
Timetable_Optimisation_Recommendations.Timetable_Evaluator.Weights,
135
WriteToCache
Timetable_Optimisation_Recommendations.Bus_Operator_API_Interface.CacheWriter,
30

X
Timetable_Optimisation_Recommendations.Windows.Highlight,
44