

# Reading Buses API

V2.1.0

Generated by Doxygen 1.8.18



<b>1 Main Page</b>	<b>1</b>
1.1 Reading Buses API	1
1.1.1 Examples	1
1.1.2 Documentation	1
1.1.3 Download & Installation	1
1.2 Very Quick Start	1
1.2.0.1 Get a List of Bus Services	2
1.2.0.2 Get a List of Locations (Bus Stops)	2
1.2.0.3 Get GPS Data	2
<b>2 Namespace Index</b>	<b>3</b>
2.1 Packages	3
<b>3 Hierarchical Index</b>	<b>5</b>
3.1 Class Hierarchy	5
<b>4 Class Index</b>	<b>7</b>
4.1 Class List	7
<b>5 File Index</b>	<b>9</b>
5.1 File List	9
<b>6 Namespace Documentation</b>	<b>11</b>
6.1 ReadingBusesAPI Namespace Reference	11
6.2 ReadingBusesAPI.BusServices Namespace Reference	11
6.3 ReadingBusesAPI.BusStops Namespace Reference	11
6.4 ReadingBusesAPI.Common Namespace Reference	12
6.4.1 Enumeration Type Documentation	12
6.4.1.1 Company	12
6.4.1.2 Direction	12
6.5 ReadingBusesAPI.ErrorManagement Namespace Reference	13
6.6 ReadingBusesAPI.JourneyDetails Namespace Reference	13
6.7 ReadingBusesAPI.TimeTable Namespace Reference	13
6.8 ReadingBusesAPI.VehiclePositions Namespace Reference	14
<b>7 Class Documentation</b>	<b>15</b>
7.1 ReadingBusesAPI.TimeTable.ArchivedBusTimeTable Class Reference	15
7.1.1 Detailed Description	16
7.1.2 Constructor & Destructor Documentation	16
7.1.2.1 ArchivedBusTimeTable()	16
7.1.3 Member Function Documentation	16
7.1.3.1 ArrivalLateness()	16
7.1.3.2 DepartureLateness()	16
7.1.3.3 GetGroupedTimeTable()	17

7.1.3.4 GetTimeTable()	17
7.1.4 Property Documentation	18
7.1.4.1 ActArrivalTime	18
7.1.4.2 ActDepartureTime	18
7.2 ReadingBusesAPI.VehiclePositions.ArchivedPositions Class Reference	18
7.2.1 Detailed Description	19
7.2.2 Constructor & Destructor Documentation	19
7.2.2.1 ArchivedPositions()	19
7.2.3 Member Function Documentation	19
7.2.3.1 GetPoint()	20
7.2.4 Property Documentation	20
7.2.4.1 Latitude	20
7.2.4.2 Longitude	20
7.2.4.3 Observed	20
7.2.4.4 OperatorCode	20
7.2.4.5 Vehicle	21
7.3 ReadingBusesAPI.BusServices.BusService Class Reference	21
7.3.1 Detailed Description	22
7.3.2 Constructor & Destructor Documentation	22
7.3.2.1 BusService() [1/3]	22
7.3.2.2 BusService() [2/3]	22
7.3.2.3 BusService() [3/3]	23
7.3.3 Member Function Documentation	23
7.3.3.1 GetArchivedTimeTable() [1/2]	23
7.3.3.2 GetArchivedTimeTable() [2/2]	24
7.3.3.3 GetGroupedArchivedTimeTable() [1/2]	24
7.3.3.4 GetGroupedArchivedTimeTable() [2/2]	25
7.3.3.5 GetGroupedTimeTable() [1/2]	26
7.3.3.6 GetGroupedTimeTable() [2/2]	26
7.3.3.7 GetLivePositions()	26
7.3.3.8 GetLocations()	27
7.3.3.9 GetLocationsActo()	27
7.3.3.10 GetStops()	27
7.3.3.11 GetTimeTable() [1/2]	27
7.3.3.12 GetTimeTable() [2/2]	28
7.3.4 Member Data Documentation	28
7.3.4.1 _stops	28
7.3.4.2 _stopsObjects	29
7.3.5 Property Documentation	29
7.3.5.1 BrandName	29
7.3.5.2 OperatorCode	29
7.3.5.3 ServiceId	29

7.4 ReadingBusesAPI.BusStops.BusStop Class Reference	29
7.4.1 Detailed Description	30
7.4.2 Constructor & Destructor Documentation	30
7.4.2.1 BusStop() [1/2]	31
7.4.2.2 BusStop() [2/2]	31
7.4.3 Member Function Documentation	31
7.4.3.1 GetArchivedTimeTable() [1/2]	31
7.4.3.2 GetArchivedTimeTable() [2/2]	31
7.4.3.3 GetLiveData()	33
7.4.3.4 GetPoint()	33
7.4.3.5 GetServices()	33
7.4.3.6 GetTimeTable() [1/2]	34
7.4.3.7 GetTimeTable() [2/2]	34
7.4.4 Property Documentation	35
7.4.4.1 ActoCode	35
7.4.4.2 Bearing	35
7.4.4.3 CommonName	35
7.4.4.4 GroupName	35
7.4.4.5 Latitude	35
7.4.4.6 Longitude	36
7.4.4.7 ServicesString	36
7.5 ReadingBusesAPI.TimeTable.BusTimeTable Class Reference	36
7.5.1 Detailed Description	37
7.5.2 Constructor & Destructor Documentation	37
7.5.2.1 BusTimeTable()	37
7.5.3 Member Function Documentation	37
7.5.3.1 GetGroupedTimeTable()	37
7.5.3.2 GetTimeTable()	38
7.6 ReadingBusesAPI.ErrorManagement.ErrorFormat Class Reference	38
7.6.1 Detailed Description	39
7.6.2 Property Documentation	39
7.6.2.1 Code	39
7.6.2.2 Message	39
7.6.2.3 Status	39
7.7 ReadingBusesAPI.ErrorManagement.ErrorManager Class Reference	39
7.7.1 Detailed Description	40
7.7.2 Member Function Documentation	40
7.7.2.1 TryErrorMessageRetrieval()	40
7.8 ReadingBusesAPI.VehiclePositions.GpsController Class Reference	40
7.8.1 Detailed Description	41
7.8.2 Constructor & Destructor Documentation	41
7.8.2.1 GpsController()	41

7.8.3 Member Function Documentation	41
7.8.3.1 GetArchivedVehiclePositions() [1/2]	41
7.8.3.2 GetArchivedVehiclePositions() [2/2]	42
7.8.3.3 GetLiveVehiclePosition()	43
7.8.3.4 GetLiveVehiclePositions()	43
7.8.3.5 IsCacheValid()	44
7.8.3.6 IsVehicle()	44
7.8.4 Member Data Documentation	44
7.8.4.1 _lastRetrieval	44
7.8.4.2 _livePositionCache	44
7.9 ReadingBusesAPI.VehiclePositions.LivePosition Class Reference	45
7.9.1 Detailed Description	45
7.9.2 Constructor & Destructor Documentation	45
7.9.2.1 LivePosition()	45
7.9.3 Member Function Documentation	46
7.9.3.1 GetService()	46
7.9.4 Property Documentation	46
7.9.4.1 Bearing	46
7.9.4.2 ServiceId	46
7.10 ReadingBusesAPI.JourneyDetails.LiveRecord Class Reference	46
7.10.1 Detailed Description	47
7.10.2 Constructor & Destructor Documentation	47
7.10.2.1 LiveRecord()	47
7.10.3 Member Function Documentation	47
7.10.3.1 ArrivalMin()	48
7.10.3.2 DisplayTime()	48
7.10.3.3 GetLiveData()	48
7.10.3.4 Service()	49
7.10.4 Property Documentation	49
7.10.4.1 Destination	49
7.10.4.2 ExptArrival	49
7.10.4.3 OperatorCode	49
7.10.4.4 SchArrival	49
7.10.4.5 ServiceNumber	50
7.10.4.6 VehicleRef	50
7.10.4.7 ViaMessage	50
7.11 ReadingBusesAPI.BusStops.Locations Class Reference	50
7.11.1 Detailed Description	50
7.11.2 Member Function Documentation	50
7.11.2.1 FindLocations()	50
7.11.3 Member Data Documentation	51
7.11.3.1 CacheLocation	51

7.12 ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter Class Reference . . . . .	51
7.12.1 Detailed Description . . . . .	52
7.12.2 Member Function Documentation . . . . .	52
7.12.2.1 CanConvert() . . . . .	52
7.12.2.2 ReadJson() . . . . .	52
7.12.2.3 WriteJson() . . . . .	52
7.12.3 Member Data Documentation . . . . .	52
7.12.3.1 Singleton . . . . .	52
7.13 ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter Class Reference . . . . .	53
7.13.1 Detailed Description . . . . .	53
7.13.2 Member Function Documentation . . . . .	53
7.13.2.1 CanConvert() . . . . .	53
7.13.2.2 ReadJson() . . . . .	53
7.13.2.3 WriteJson() . . . . .	54
7.13.3 Member Data Documentation . . . . .	54
7.13.3.1 Singleton . . . . .	54
7.14 ReadingBusesAPI.Common.ParseOperatorConverter Class Reference . . . . .	54
7.14.1 Detailed Description . . . . .	54
7.14.2 Member Function Documentation . . . . .	55
7.14.2.1 CanConvert() . . . . .	55
7.14.2.2 ReadJson() . . . . .	55
7.14.2.3 WriteJson() . . . . .	55
7.14.3 Member Data Documentation . . . . .	55
7.14.3.1 Singleton . . . . .	55
7.15 ReadingBusesAPI.Common.ParseStringConverter Class Reference . . . . .	56
7.15.1 Detailed Description . . . . .	56
7.15.2 Member Function Documentation . . . . .	56
7.15.2.1 CanConvert() . . . . .	56
7.15.2.2 ReadJson() . . . . .	56
7.15.2.3 WriteJson() . . . . .	57
7.15.3 Member Data Documentation . . . . .	57
7.15.3.1 Singleton . . . . .	57
7.16 ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter Class Reference . . . . .	57
7.16.1 Detailed Description . . . . .	58
7.16.2 Member Function Documentation . . . . .	58
7.16.2.1 CanConvert() . . . . .	58
7.16.2.2 ReadJson() . . . . .	58
7.16.2.3 WriteJson() . . . . .	58
7.16.3 Member Data Documentation . . . . .	58
7.16.3.1 Singleton . . . . .	58
7.17 ReadingBusesAPI.Common.Point Struct Reference . . . . .	59
7.17.1 Detailed Description . . . . .	59

7.17.2 Constructor & Destructor Documentation	59
7.17.2.1 Point()	59
7.17.3 Member Function Documentation	60
7.17.3.1 Equals() [1/2]	60
7.17.3.2 Equals() [2/2]	60
7.17.3.3 GetHashCode()	60
7.17.3.4 operator"!="()	61
7.17.3.5 operator=="()	61
7.17.3.6 ToString()	61
7.17.4 Property Documentation	62
7.17.4.1 X	62
7.17.4.2 Y	62
7.18 ReadingBusesAPI.ReadingBuses Class Reference	62
7.18.1 Detailed Description	64
7.18.2 Constructor & Destructor Documentation	65
7.18.2.1 ReadingBuses()	65
7.18.3 Member Function Documentation	65
7.18.3.1 DeleteInstance()	65
7.18.3.2 GetInstance()	65
7.18.3.3 GetLocation()	66
7.18.3.4 GetLocations()	66
7.18.3.5 GetOperatorE()	66
7.18.3.6 GetService() [1/2]	67
7.18.3.7 GetService() [2/2]	67
7.18.3.8 GetServices() [1/3]	68
7.18.3.9 GetServices() [2/3]	68
7.18.3.10 GetServices() [3/3]	68
7.18.3.11 GetVehicleTrackingHistory()	69
7.18.3.12 Initialise()	69
7.18.3.13 InvalidateCache()	70
7.18.3.14 IsLocation()	70
7.18.3.15 IsService() [1/2]	70
7.18.3.16 IsService() [2/2]	71
7.18.3.17 PrintFullErrorLogs()	71
7.18.3.18 PrintWarning()	71
7.18.3.19 SetCache()	71
7.18.3.20 SetCacheValidityLength()	72
7.18.3.21 SetDebugging()	72
7.18.3.22 SetFullError()	72
7.18.3.23 SetUp()	73
7.18.3.24 SetWarning()	73
7.18.4 Member Data Documentation	73



7.18.4.1 _instance	73
7.18.4.2 _locations	74
7.18.4.3 _services	74
7.18.5 Property Documentation	74
7.18.5.1 ApiKey	74
7.18.5.2 Cache	74
7.18.5.3 CacheValidityLength	74
7.18.5.4 Debugging	74
7.18.5.5 FullError	75
7.18.5.6 GpsController	75
7.18.5.7 Warning	75
7.19 ReadingBusesAPI.ErrorManagement.ReadingBusesApiException Class Reference	75
7.19.1 Detailed Description	75
7.19.2 Constructor & Destructor Documentation	76
7.19.2.1 ReadingBusesApiException() [1/3]	76
7.19.2.2 ReadingBusesApiException() [2/3]	76
7.19.2.3 ReadingBusesApiException() [3/3]	76
7.20 ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery Class Reference	76
7.20.1 Detailed Description	77
7.20.2 Constructor & Destructor Documentation	77
7.20.2.1 ReadingBusesApiExceptionBadQuery() [1/3]	77
7.20.2.2 ReadingBusesApiExceptionBadQuery() [2/3]	77
7.20.2.3 ReadingBusesApiExceptionBadQuery() [3/3]	77
7.21 ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical Class Reference	78
7.21.1 Detailed Description	78
7.21.2 Constructor & Destructor Documentation	78
7.21.2.1 ReadingBusesApiExceptionCritical() [1/3]	78
7.21.2.2 ReadingBusesApiExceptionCritical() [2/3]	78
7.21.2.3 ReadingBusesApiExceptionCritical() [3/3]	79
7.22 ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery Class Reference	79
7.22.1 Detailed Description	79
7.22.2 Constructor & Destructor Documentation	79
7.22.2.1 ReadingBusesApiExceptionMalformedQuery() [1/2]	80
7.22.2.2 ReadingBusesApiExceptionMalformedQuery() [2/2]	80
7.23 ReadingBusesAPI.BusServices.Services Class Reference	80
7.23.1 Detailed Description	80
7.23.2 Member Function Documentation	80
7.23.2.1 FindServices()	80
7.23.3 Member Data Documentation	81
7.23.3.1 CacheLocation	81
7.24 ReadingBusesAPI.TimeTable.TimeTableRecord Class Reference	81
7.24.1 Detailed Description	82

7.24.2 Constructor & Destructor Documentation	82
7.24.2.1 TimeTableRecord()	82
7.24.3 Member Function Documentation	82
7.24.3.1 GetService()	83
7.24.4 Property Documentation	83
7.24.4.1 Direction	83
7.24.4.2 IsTimingPoint	83
7.24.4.3 JourneyCode	83
7.24.4.4 Location	83
7.24.4.5 Operator	84
7.24.4.6 SchArrivalTime	84
7.24.4.7 SchDepartureTime	84
7.24.4.8 Sequence	84
7.24.4.9 ServiceNumber	84
7.25 ReadingBusesAPI.Common.UrlConstructor Class Reference	84
7.25.1 Detailed Description	85
7.25.2 Member Function Documentation	85
7.25.2.1 AddTimeSpan()	85
7.25.2.2 LinePatterns()	86
7.25.2.3 ListOfBusStops()	86
7.25.2.4 ListOfServices()	86
7.25.2.5 LiveJourneyDetails()	87
7.25.2.6 LiveVehiclePositions()	87
7.25.2.7 StopPredictions()	87
7.25.2.8 TimetabledJourneys()	87
7.25.2.9 TrackingHistory()	88
7.25.2.10 VehiclePositionHistory()	88
7.25.3 Member Data Documentation	89
7.25.3.1 DummyApi	89
7.25.3.2 ReadingBusesApi	89
<b>8 File Documentation</b>	<b>91</b>
8.1 BusServices/BusService.cs File Reference	91
8.2 BusServices/Services.cs File Reference	91
8.3 BusStops/BusStop.cs File Reference	91
8.4 BusStops/Locations.cs File Reference	92
8.5 Common/Direction.cs File Reference	92
8.6 Common/JSONConverters.cs File Reference	92
8.7 Common/Operators.cs File Reference	93
8.8 Common/Point.cs File Reference	93
8.9 Common/URLConstructor.cs File Reference	93
8.10 ErrorManagement/ErrorFormat.cs File Reference	93

---

8.11 ErrorManagement/ErrorManager.cs File Reference . . . . .	94
8.12 ErrorManagement/ReadingBusesAPIException.cs File Reference . . . . .	94
8.13 ErrorManagement/ReadingBusesApiExceptionBadQuery.cs File Reference . . . . .	94
8.14 ErrorManagement/ReadingBusesApiExceptionCritical.cs File Reference . . . . .	95
8.15 ErrorManagement/ReadingBusesApiExceptionMalformedQuery.cs File Reference . . . . .	95
8.16 JourneyDetails/LiveRecord.cs File Reference . . . . .	95
8.17 ReadingBuses.cs File Reference . . . . .	96
8.18 README.md File Reference . . . . .	96
8.19 TimeTable/ArchivedBusTimeTable.cs File Reference . . . . .	96
8.20 TimeTable/BusTimeTable.cs File Reference . . . . .	96
8.21 TimeTable/TimeTableRecord.cs File Reference . . . . .	97
8.22 VehiclePositions/ArchivedPositions.cs File Reference . . . . .	97
8.23 VehiclePositions/GPSController.cs File Reference . . . . .	97
8.24 VehiclePositions/LivePosition.cs File Reference . . . . .	97
<b>Index</b>	<b>99</b>



# Chapter 1

## Main Page

### 1.1 Reading Buses API

A C#, .net Standard Library for the [Reading Buses API](#), available to use in your C# console, UWP, WPF or Win Form Applications.

The library supports the List of Bus Stops, Live Vehicle Positions, Stop Predictions, List of Services, Line Patterns, Timetabled Journeys, Tracking History and the Vehicle Position History API..

Get your own API Keys from: <http://rtl2.ods-live.co.uk/cms/apiservice>

#### 1.1.1 Examples

Examples code can be found on the [Examples Repository here](#)

#### 1.1.2 Documentation

Documentation can be found by [online here](#) or the "docs" folder of this repository.

#### 1.1.3 Download & Installation

Get the package from the [nuget store here](#)

### 1.2 Very Quick Start

First you need to initialise the Library by providing your API key:

```
{c#}  
ReadingBuses Controller = await ReadingBuses.Initialise("APIKEY");
```

Once Initilised all future reference can be got using:

```
{c#}  
ReadingBuses Controller = ReadingBuses.GetInstance();
```

### 1.2.0.1 Get a List of Bus Services

To get a list of bus services operated by Reading Buses and the information about them you can do any of the following:

***To get all Services:***

```
{c#}  
BusService[] Services = Controller.GetServices();
```

***To get all Services from a company:***

```
{c#}  
BusService[] RBServices = Controller.GetServices(Company.ReadingBuses);  
BusService[] KNServices = Controller.GetServices(Company.Kennections);
```

***To get specific Services based on their brand:***

```
{c#}  
BusService[] Services = Controller.GetServices("pink");
```

***To get specific Service by Service Number and Company Operator:***

```
{c#}  
BusService Services = Controller.GetService("17", Company.ReadingBuses);
```

### 1.2.0.2 Get a List of Locations (Bus Stops)

***To get all locations***

```
{c#}  
BusStop[] Locations = Controller.GetLocations();
```

***To get location based on Acto-Code (Bus Stop ID)***

```
{c#}  
BusStop Locations = Controller.GetLocation("33245365434");
```

### 1.2.0.3 Get GPS Data

***To get Live GPS Data***

```
{c#}  
LivePosition[] Positions = await Controller.GpsController.GetLiveVehiclePositions();
```

***To get Archived GPS Data***

```
{c#}  
LivePosition[] Positions = await  
    Controller.GpsController.GetArchivedVehiclePositions(DateTime.Now.AddDays(-1), new TimeSpan(3, 0,  
        0));
```

Once you have the bus service or location you want to inspect, they have various properties to let you get further data about them. Such as live bus stop data, GPS data, bus service routes. For more examples please see the repository linked above.

## Chapter 2

# Namespace Index

### 2.1 Packages

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

<a href="#">ReadingBusesAPI</a>	11
<a href="#">ReadingBusesAPI.BusServices</a>	11
<a href="#">ReadingBusesAPI.BusStops</a>	11
<a href="#">ReadingBusesAPI.Common</a>	12
<a href="#">ReadingBusesAPI.ErrorManagement</a>	13
<a href="#">ReadingBusesAPI.JourneyDetails</a>	13
<a href="#">ReadingBusesAPI.TimeTable</a>	13
<a href="#">ReadingBusesAPI.VehiclePositions</a>	14





## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

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

ReadingBusesAPI.VehiclePositions.ArchivedPositions . . . . .	18
ReadingBusesAPI.VehiclePositions.LivePosition . . . . .	45
ReadingBusesAPI.BusServices.BusService . . . . .	21
ReadingBusesAPI.BusStops.BusStop . . . . .	29
ReadingBusesAPI.ErrorManagement.ErrorFormat . . . . .	38
ReadingBusesAPI.ErrorManagement.ErrorManager . . . . .	39
Exception	
ReadingBusesAPI.ErrorManagement.ReadingBusesApiException . . . . .	75
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery . . . . .	76
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical . . . . .	78
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery . . . . .	79
ReadingBusesAPI.VehiclePositions.GpsController . . . . .	40
IEquatable	
ReadingBusesAPI.Common.Point . . . . .	59
JsonConverter	
ReadingBusesAPI.Common.ParseOperatorConverter . . . . .	54
ReadingBusesAPI.Common.ParseStringConverter . . . . .	56
ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter . . . . .	51
ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter . . . . .	53
ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter . . . . .	57
ReadingBusesAPI.JourneyDetails.LiveRecord . . . . .	46
ReadingBusesAPI.BusStops.Locations . . . . .	50
ReadingBusesAPI.ReadingBuses . . . . .	62
ReadingBusesAPI.BusServices.Services . . . . .	80
ReadingBusesAPI.TimeTable.TimeTableRecord . . . . .	81
ReadingBusesAPI.TimeTable.ArchivedBusTimeTable . . . . .	15
ReadingBusesAPI.TimeTable.BusTimeTable . . . . .	36
ReadingBusesAPI.Common.UrlConstructor . . . . .	84



## Chapter 4

# Class Index

### 4.1 Class List

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

<a href="#">ReadingBusesAPI.TimeTable.ArchivedBusTimeTable</a>	
Represents and retrieves information about a actual single time table record, which means information on one bus at one location. Related to the "Tracking History" API . . . . .	15
<a href="#">ReadingBusesAPI.VehiclePositions.ArchivedPositions</a>	
Stores information about previous/ archived GPS data on vehicles . . . . .	18
<a href="#">ReadingBusesAPI.BusServices.BusService</a>	
Stores information about an individual bus services. Related to the "List Of Services" API . . .	21
<a href="#">ReadingBusesAPI.BusStops.BusStop</a>	
Stores information about a single bus stop. Related to the "List Of Bus Stops" API . . . . .	29
<a href="#">ReadingBusesAPI.TimeTable.BusTimeTable</a>	
Represents and retrieves information about a scheduled/predicted single time table record, which means information on one bus at one location. Related to the "Timetabled Journeys" API . . .	36
<a href="#">ReadingBusesAPI.ErrorManagement.ErrorFormat</a>	
Represents an error message object returned by all JSON feeds of the API . . . . .	38
<a href="#">ReadingBusesAPI.ErrorManagement.ErrorManager</a>	
Responsible for extracting and producing an error message from an API result. To the end user	39
<a href="#">ReadingBusesAPI.VehiclePositions.GpsController</a>	
Helps get live and historical GPS data on vehicles by accessing the "Live Vehicle Positions" API	40
<a href="#">ReadingBusesAPI.VehiclePositions.LivePosition</a>	
Used to store live information about a buses GPS position. Related to the "Live Vehicle Positions" API . . . . .	45
<a href="#">ReadingBusesAPI.JourneyDetails.LiveRecord</a>	
Used to store information about a buses arrival at a bus stop. Mainly related to the "Stop Predictions" API . . . . .	46
<a href="#">ReadingBusesAPI.BusStops.Locations</a>	
This classes simply gets all the buses stops visited by Reading Buses, by interfacing with the "List Of Bus Stops" API . . . . .	50
<a href="#">ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter</a>	
Converts a bus stop acto-code into a 'BusStop' Object and back again for the JSON converter	51
<a href="#">ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter</a>	
Converts a string into a Direction Enum and back again for the JSON converter . . . . .	53
<a href="#">ReadingBusesAPI.Common.ParseOperatorConverter</a>	
Converts a string short code for an Operator into an Operator Enum and back again for the JSON converter . . . . .	54
<a href="#">ReadingBusesAPI.Common.ParseStringConverter</a>	
Converts a string into a long and back again for the JSON converter . . . . .	56

<a href="#">ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter</a>	
Converts a string into a boolean and back again for the JSON converter . . . . .	57
<a href="#">ReadingBusesAPI.Common.Point</a>	
Stores an X and Y Position simply . . . . .	59
<a href="#">ReadingBusesAPI.ReadingBuses</a>	
This is the main class for the library, here you can initialise a singleton instance and then query and use the Reading Buses API . . . . .	62
<a href="#">ReadingBusesAPI.ErrorManagement.ReadingBusesApiException</a>	
Stores the basic/base type of Exception which can be thrown by the API . . . . .	75
<a href="#">ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery</a>	
An exception type which is used when the API returns back an error message. Most likely due to an invalid request such as asking for data that does not exist . . . . .	76
<a href="#">ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical</a>	
An exception type which is thrown when the cause of the error is unknown . . . . .	78
<a href="#">ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery</a>	
An exception type which is used when the user asks to make a invalid API call This is would be thrown during checks done before even directly calling upon the web API. For example if you have not filtered by at least one property when required too . . . . .	79
<a href="#">ReadingBusesAPI.BusServices.Services</a>	
This classes simply gets all the bus services operated by Reading Buses, by interfacing with the "List Of Services" API . . . . .	80
<a href="#">ReadingBusesAPI.TimeTable.TimeTableRecord</a>	
Represents the Raw timetable object data you get from the Timetabled Journeys and Tracking History APIs . . . . .	81
<a href="#">ReadingBusesAPI.Common.UrlConstructor</a>	
Returns back the URL needed to make a get command to the Reading Buses Open Data API. You can use this for testing purposes to check the API is returning what you were expecting . .	84

## Chapter 5

# File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

<a href="#">ReadingBuses.cs</a>	96
<a href="#">BusServices/BusService.cs</a>	91
<a href="#">BusServices/Services.cs</a>	91
<a href="#">BusStops/BusStop.cs</a>	91
<a href="#">BusStops/Locations.cs</a>	92
<a href="#">Common/Direction.cs</a>	92
<a href="#">Common/JSONConverters.cs</a>	92
<a href="#">Common/Operators.cs</a>	93
<a href="#">Common/Point.cs</a>	93
<a href="#">Common/URLConstructor.cs</a>	93
<a href="#">ErrorManagement/ErrorFormat.cs</a>	93
<a href="#">ErrorManagement/ErrorHandler.cs</a>	94
<a href="#">ErrorManagement/ReadingBusesAPIException.cs</a>	94
<a href="#">ErrorManagement/ReadingBusesApiExceptionBadQuery.cs</a>	94
<a href="#">ErrorManagement/ReadingBusesApiExceptionCritical.cs</a>	95
<a href="#">ErrorManagement/ReadingBusesApiExceptionMalformedQuery.cs</a>	95
<a href="#">JourneyDetails/LiveRecord.cs</a>	95
<a href="#">TimeTable/ArchivedBusTimeTable.cs</a>	96
<a href="#">TimeTable/BusTimeTable.cs</a>	96
<a href="#">TimeTable/TimeTableRecord.cs</a>	97
<a href="#">VehiclePositions/ArchivedPositions.cs</a>	97
<a href="#">VehiclePositions/GPSController.cs</a>	97
<a href="#">VehiclePositions/LivePosition.cs</a>	97



## Chapter 6

# Namespace Documentation

### 6.1 ReadingBusesAPI Namespace Reference

#### Namespaces

- namespace [BusServices](#)
- namespace [BusStops](#)
- namespace [Common](#)
- namespace [ErrorManagement](#)
- namespace [JourneyDetails](#)
- namespace [TimeTable](#)
- namespace [VehiclePositions](#)

#### Classes

- class [ReadingBuses](#)

*This is the main class for the library, here you can initialise a singleton instance and then query and use the Reading Buses API.*

### 6.2 ReadingBusesAPI.BusServices Namespace Reference

#### Classes

- class [BusService](#)

*Stores information about an individual bus services. Related to the "List Of Services" API.*

- class [Services](#)

*This classes simply gets all the bus services operated by Reading Buses, by interfacing with the "List Of Services" API.*

### 6.3 ReadingBusesAPI.BusStops Namespace Reference

#### Classes

- class [BusStop](#)

*Stores information about a single bus stop. Related to the "List Of Bus Stops" API.*

- class [Locations](#)

*This classes simply gets all the buses stops visited by Reading Buses, by interfacing with the "List Of Bus Stops" API.*

## 6.4 ReadingBusesAPI.Common Namespace Reference

### Classes

- class [ParseOperatorConverter](#)  
*Converts a string short code for an Operator into an Operator Enum and back again for the JSON converter.*
- class [ParseStringConverter](#)  
*Converts a string into a long and back again for the JSON converter.*
- struct [Point](#)  
*Stores an X and Y Position simply.*
- class [UrlConstructor](#)  
*Returns back the URL needed to make a get command to the Reading Buses Open Data API. You can use this for testing purposes to check the API is returning what you were expecting.*

### Enumerations

- enum [Direction](#) { [Direction.Inbound](#), [Direction.Outbound](#) }  
*The direction in which a bus is traveling.*
- enum [Company](#) { [Company.ReadingBuses](#), [Company.Kennections](#), [Company.NewburyAndDistrict](#), [Company.Other](#) }  
*An Enum of the Operators Reading Buses owns or manages in their API.*

### 6.4.1 Enumeration Type Documentation

#### 6.4.1.1 Company

```
enum ReadingBusesAPI.Common.Company [strong]
```

An Enum of the Operators Reading Buses owns or manages in their API.

##### Enumerator

<a href="#">ReadingBuses</a>	For Reading Buses services
<a href="#">Kennections</a>	For Kennections services
<a href="#">NewburyAndDistrict</a>	For Newbury And District services
<a href="#">Other</a>	For any other operator which is new in the API and has not yet been officially supported in this library.

#### 6.4.1.2 Direction

```
enum ReadingBusesAPI.Common.Direction [strong]
```

The direction in which a bus is traveling.



## Enumerator

Inbound	For buses traveling inbound.
Outbound	For buses traveling outbound.

## 6.5 ReadingBusesAPI.ErrorManagement Namespace Reference

### Classes

- class [ErrorFormat](#)  
*Represents an error message object returned by all JSON feeds of the API.*
- class [ErrorManager](#)  
*Responsible for extracting and producing an error message from an API result. To the end user.*
- class [ReadingBusesApiException](#)  
*Stores the basic/base type of Exception which can be thrown by the API.*
- class [ReadingBusesApiExceptionBadQuery](#)  
*An exception type which is used when the API returns back an error message. Most likely due to an invalid request such as asking for data that does not exist.*
- class [ReadingBusesApiExceptionCritical](#)  
*An exception type which is thrown when the cause of the error is unknown.*
- class [ReadingBusesApiExceptionMalformedQuery](#)  
*An exception type which is used when the user asks to make a invalid API call This is would be thrown during checks done before even directly calling upon the web API. For example if you have not filtered by at least one property when required too.*

## 6.6 ReadingBusesAPI.JourneyDetails Namespace Reference

### Classes

- class [LiveRecord](#)  
*Used to store information about a buses arrival at a bus stop. Mainly related to the "Stop Predictions" API.*

## 6.7 ReadingBusesAPI.TimeTable Namespace Reference

### Classes

- class [ArchivedBusTimeTable](#)  
*Represents and retrieves information about a actual single time table record, which means information on one bus at one location. Related to the "Tracking History" API.*
- class [BusTimeTable](#)  
*Represents and retrieves information about a scheduled/predicted single time table record, which means information on one bus at one location. Related to the "Timetabled Journeys" API.*
- class [TimeTableRecord](#)  
*Represents the Raw timetable object data you get from the Timetabled Journeys and Tracking History APIs.*

## 6.8 ReadingBusesAPI.VehiclePositions Namespace Reference

### Classes

- class [ArchivedPositions](#)

*Stores information about previous/ archived GPS data on vehicles.*

- class [GpsController](#)

*Helps get live and historical GPS data on vehicles by accessing the "Live Vehicle Positions" API.*

- class [LivePosition](#)

*Used to store live information about a buses GPS position. Related to the "Live Vehicle Positions" API.*

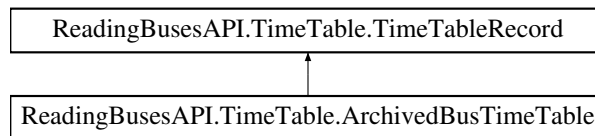
## Chapter 7

# Class Documentation

### 7.1 ReadingBusesAPI.TimeTable.ArchivedBusTimeTable Class Reference

Represents and retrieves information about a actual single time table record, which means information on one bus at one location. Related to the "Tracking History" API.

Inheritance diagram for ReadingBusesAPI.TimeTable.ArchivedBusTimeTable:



#### Public Member Functions

- double [ArrivalLateness](#) ()  
*How late the bus was to arrive at a bus stop.*
- double [DepartureLateness](#) ()  
*How late the bus was to departure at a bus stop.*

#### Package Functions

- [ArchivedBusTimeTable](#) ()  
*Default constructor to prevent creating an object directly outside the API.*

#### Static Package Functions

- static async Task< [ArchivedBusTimeTable](#)[]> [GetTimeTable](#) ([BusService](#) service, DateTime date, [BusStop](#) location, string vehicle)  
*Gets the actual arrival and departure times of a bus, by service, date, location and/or vehicle ID.*
- static async Task< IGrouping< string, [ArchivedBusTimeTable](#) >[]> [GetGroupedTimeTable](#) ([BusService](#) service, DateTime date, [BusStop](#) location, string vehicle)  
*Gets the time table for a service and groups it by a journey code instead of one continuous array of time table entries.*

## Properties

- DateTime? [ActArrivalTime](#) [get, set]  
*The actual arrival time for the bus.*
- DateTime? [ActDepartureTime](#) [get, set]  
*The actual departure time for the bus.*

### 7.1.1 Detailed Description

Represents and retrieves information about a actual single time table record, which means information on one bus at one location. Related to the "Tracking History" API.

### 7.1.2 Constructor & Destructor Documentation

#### 7.1.2.1 ArchivedBusTimeTable()

```
ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.ArchivedBusTimeTable ( ) [package]
```

Default constructor to prevent creating an object directly outside the API.

### 7.1.3 Member Function Documentation

#### 7.1.3.1 Arrivallateness()

```
double ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.Arrivallateness ( )
```

How late the bus was to arrive at a bus stop.

##### Returns

The number of seconds the bus was late to arrive by. If no arrival time can be found, 0 is returned.

#### 7.1.3.2 DepartureLateness()

```
double ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.DepartureLateness ( )
```

How late the bus was to departure at a bus stop.

##### Returns

The number of seconds the bus was late to departure by. If no departure time can be found, 0 is returned.

### 7.1.3.3 GetGroupedTimeTable()

```
static async Task<IGrouping<string, ArchivedBusTimeTable>[]> ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.GetGroupedTimeTable (
    BusService service,
    DateTime date,
    BusStop location,
    string vehicle ) [static], [package]
```

Gets the time table for a service and groups it by a journey code instead of one continuous array of time table entries.

#### Parameters

<i>service</i>	The bus services you wish to view.
<i>date</i>	The date of the time table.
<i>location</i>	The location to get timetable data from.
<i>vehicle</i>	A bus/Vehicle ID number.

#### Returns

Returns an IGroupings of Arrays of 'BusTimeTable' records grouped by journey codes.

#### Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

### 7.1.3.4 GetTimeTable()

```
static async Task<ArchivedBusTimeTable[]> ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.GetTimeTable (
    BusService service,
    DateTime date,
    BusStop location,
    string vehicle ) [static], [package]
```

Gets the actual arrival and departure times of a bus, by service, date, location and/or vehicle ID.

#### Parameters

<i>service</i>	The bus services you wish to view.
<i>date</i>	The date of the time table.
<i>location</i>	The location to get timetable data from.
<i>vehicle</i>	A bus/Vehicle ID number.

**Returns**

An array of time table records for the service or location or both

**Exceptions**

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

See also [BusTimeTable.GetTimeTable\(BusService , DateTime ,BusStop\)](#) to get future time table data instead.

**7.1.4 Property Documentation****7.1.4.1 ActArrivalTime**

`DateTime? ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.ActArrivalTime [get], [set]`

The actual arrival time for the bus.

**7.1.4.2 ActDepartureTime**

`DateTime? ReadingBusesAPI.TimeTable.ArchivedBusTimeTable.ActDepartureTime [get], [set]`

The actual departure time for the bus.

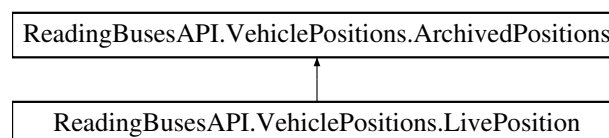
The documentation for this class was generated from the following file:

- TimeTable/[ArchivedBusTimeTable.cs](#)

**7.2 ReadingBusesAPI.VehiclePositions.ArchivedPositions Class Reference**

Stores information about previous/ archived GPS data on vehicles.

Inheritance diagram for ReadingBusesAPI.VehiclePositions.ArchivedPositions:



## Public Member Functions

- [Point GetPoint](#) ()  
*Gets the geographical position of the bus.*

## Package Functions

- [ArchivedPositions](#) ()  
*The default constructor, which sets the 'LastRetrieval' to current time.*

## Properties

- [Company OperatorCode](#) [get, set]  
*Holds the operators enum value.*
- string [Vehicle](#) [get, set]  
*Holds the reference/identifier for the vehicle*
- DateTimeOffset [Observed](#) [get, set]  
*Holds the time it was last seen/ new data was retrieved.*
- string [Latitude](#) [get, set]  
*Latitude position of the bus*
- string [Longitude](#) [get, set]  
*longitude position of the bus*

### 7.2.1 Detailed Description

Stores information about previous/ archived GPS data on vehicles.

### 7.2.2 Constructor & Destructor Documentation

#### 7.2.2.1 ArchivedPositions()

```
ReadingBusesAPI.VehiclePositions.ArchivedPositions.ArchivedPositions ( ) [package]
```

The default constructor, which sets the 'LastRetrieval' to current time.

### 7.2.3 Member Function Documentation

### 7.2.3.1 GetPoint()

```
Point ReadingBusesAPI.VehiclePositions.ArchivedPositions.GetPoint ( )
```

Gets the geographical position of the bus.

#### Returns

A Point Object for the position of the bus.

## 7.2.4 Property Documentation

### 7.2.4.1 Latitude

```
string ReadingBusesAPI.VehiclePositions.ArchivedPositions.Latitude [get], [set]
```

Latitude position of the bus

### 7.2.4.2 Longitude

```
string ReadingBusesAPI.VehiclePositions.ArchivedPositions.Longitude [get], [set]
```

longitude position of the bus

### 7.2.4.3 Observed

```
DateTimeOffset ReadingBusesAPI.VehiclePositions.ArchivedPositions.Observed [get], [set]
```

Holds the time it was last seen/ new data was retrieved.

### 7.2.4.4 OperatorCode

```
Company ReadingBusesAPI.VehiclePositions.ArchivedPositions.OperatorCode [get], [set]
```

Holds the operators enum value.



#### 7.2.4.5 Vehicle

```
string ReadingBusesAPI.VehiclePositions.ArchivedPositions.Vehicle [get], [set]
```

Holds the reference/identifier for the vehicle

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

- VehiclePositions/[ArchivedPositions.cs](#)

## 7.3 ReadingBusesAPI.BusServices.BusService Class Reference

Stores information about an individual bus services. Related to the "List Of Services" API.

### Public Member Functions

- [BusService](#) (string serviceNumber)  
*Used to create a snub/ fake object for passing to function calls, if all you need to pass is an service number to the function. Makes operator code, "other" by default.*
- [BusService](#) (string serviceNumber, [Company](#) operators)  
*Used to create a snub/ fake object for passing to function calls, if all you need to pass is an service number to the function.*
- async Task< string[]> [GetLocationsActo](#) ()  
*Gets an array of stops the bus service travels too as an array of ActoCode*
- async Task< [BusStop](#)[]> [GetLocations](#) ()  
*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.*
- async Task< [LivePosition](#)[]> [GetLivePositions](#) ()  
*Gets the Live GPS positions for all Vehicles operating on this service.*
- Task< [BusTimeTable](#)[]> [GetTimeTable](#) (DateTime date)  
*Gets the full bus time table, for a specific date.*
- Task< [BusTimeTable](#)[]> [GetTimeTable](#) (DateTime date, [BusStop](#) location)  
*Gets the full bus time table, for a specific date.*
- Task< IGrouping< string, [BusTimeTable](#) >[]> [GetGroupedTimeTable](#) (DateTime date)  
*Gets the time table for this specific bus service, split into groups by the journey code.*
- Task< IGrouping< string, [BusTimeTable](#) >[]> [GetGroupedTimeTable](#) (DateTime date, [BusStop](#) location)  
*Gets the time table for this specific bus service, split into groups by the journey code.*
- Task< [ArchivedBusTimeTable](#)[]> [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.*
- Task< [ArchivedBusTimeTable](#)[]> [GetArchivedTimeTable](#) (DateTime date, [BusStop](#) location)  
*Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date.*
- Task< IGrouping< string, [ArchivedBusTimeTable](#) >[]> [GetGroupedArchivedTimeTable](#) (DateTime date)  
*Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date, split into groups by the journey code.*
- Task< IGrouping< string, [ArchivedBusTimeTable](#) >[]> [GetGroupedArchivedTimeTable](#) (DateTime date, [BusStop](#) location)  
*Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date, split into groups by the journey code.*

## Package Functions

- [BusService](#) ()  
*The default constructor, used for automatic phrasing of data.*

## Properties

- string [ServiceId](#) [get, set]
- string [BrandName](#) [get, set]  
*The brand name for the service, used mainly for Reading Buses services, such as Lion, Purple or Orange.*
- [Company OperatorCode](#) [get, set]  
*The operator enum value.*

## Private Member Functions

- async Task< List< string > > [GetStops](#) ()  
*Gets a list of bus stops actio codes, if this is the first time it's asked for call upon the API This is delayed so only to call the API when needed.*

## Private Attributes

- List< string > [\\_stops](#)  
*Stores a list of bus stops actio-code/IDs the service visits.*
- [BusStop\[\]](#) [\\_stopsObjects](#)  
*Stores a list of 'BusStops' objects the service visits.*

### 7.3.1 Detailed Description

Stores information about an individual bus services. Related to the "List Of Services" API.

### 7.3.2 Constructor & Destructor Documentation

#### 7.3.2.1 BusService() [1/3]

```
ReadingBusesAPI.BusServices.BusService.BusService ( ) [package]
```

The default constructor, used for automatic phrasing of data.

#### 7.3.2.2 BusService() [2/3]

```
ReadingBusesAPI.BusServices.BusService.BusService (
    string serviceNumber )
```

Used to create a snub/ fake object for passing to function calls, if all you need to pass is an service number to the function. Makes operator code, "other" by default.

## Parameters

<i>serviceNumber</i>	ID of the bus service.
----------------------	------------------------

Unless you are doing something very strange, you probably should not need to use this, it is more for testing purposes.

**7.3.2.3 BusService()** [3/3]

```
ReadingBusesAPI.BusServices.BusService.BusService (
    string serviceNumber,
    Company operators )
```

Used to create a snub/ fake object for passing to function calls, if all you need to pass is an service number to the function.

## Parameters

<i>serviceNumber</i>	ID of the bus service.
<i>operators</i>	The operator who runs the service.

Unless you are doing something very strange, you probably should not need to use this, it is more for testing purposes.

**7.3.3 Member Function Documentation****7.3.3.1 GetArchivedTimeTable()** [1/2]

```
Task<ArchivedBusTimeTable[ ]> ReadingBusesAPI.BusServices.BusService.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

<i>date</i>	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.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

## 7.3.3.2 GetArchivedTimeTable() [2/2]

```
Task<ArchivedBusTimeTable[ ]> ReadingBusesAPI.BusServices.BusService.GetArchivedTimeTable (
    DateTime date,
    BusStop location )
```

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

## Parameters

<i>date</i>	the date on which you want a archived timetable data for. This should be a date in the past.
<i>location</i>	(optional) a specific bus stop you want archived timetables for, if null it will get a timetable for every bus stop on route.

## Returns

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

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

## 7.3.3.3 GetGroupedArchivedTimeTable() [1/2]

```
Task<IGrouping<string, ArchivedBusTimeTable>[ ]> ReadingBusesAPI.BusServices.BusService.Get←
GroupedArchivedTimeTable (
    DateTime date )
```

Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date, split into groups by the journey code.

## Parameters

<i>date</i>	The date on which you want the time table for. This should be a date in the past.
-------------	---

## Returns

A grouping of arrays of time table records based upon journey code.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

## 7.3.3.4 GetGroupedArchivedTimeTable() [2/2]

```
Task<IGrouping<string, ArchivedBusTimeTable>[]> ReadingBusesAPI.BusServices.BusService.Get↔
GroupedArchivedTimeTable (
    DateTime date,
    BusStop location )
```

Gets the archived real bus departure and arrival times along with their time table history for this service on a specific date, split into groups by the journey code.

## Parameters

<i>date</i>	The date on which you want the time table for. This should be a date in the past.
<i>location</i>	(optional) The specific bus stop you want time table data for. Leave as null if you want the whole routes timetable.

## Returns

A grouping of arrays of time table records based upon journey code.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

### 7.3.3.5 GetGroupedTimeTable() [1/2]

```
Task<IGrouping<string, BusTimeTable>[]> ReadingBusesAPI.BusServices.BusService.GetGroupedTimeTable (
    DateTime date )
```

Gets the time table for this specific bus service, split into groups by the journey code.

#### Parameters

<i>date</i>	The date on which you want the time table for.
-------------	--

#### Returns

A grouping of arrays of time table records based upon journey code.

### 7.3.3.6 GetGroupedTimeTable() [2/2]

```
Task<IGrouping<string, BusTimeTable>[]> ReadingBusesAPI.BusServices.BusService.GetGroupedTimeTable (
    DateTime date,
    BusStop location )
```

Gets the time table for this specific bus service, split into groups by the journey code.

#### Parameters

<i>date</i>	The date on which you want the time table for.
<i>location</i>	(optional) The specific bus stop you want time table data for. Leave as null if you want the whole routes timetable.

#### Returns

A grouping of arrays of time table records based upon journey code.

### 7.3.3.7 GetLivePositions()

```
async Task<LivePosition[]> ReadingBusesAPI.BusServices.BusService.GetLivePositions ( )
```

Gets the Live GPS positions for all Vehicles operating on this service.

#### Returns

An array of GPS data points for all vehicles currently operating on this service.

### 7.3.3.8 GetLocations()

```
async Task<BusStop[]> ReadingBusesAPI.BusServices.BusService.GetLocations ( )
```

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.

#### Returns

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

### 7.3.3.9 GetLocationsActo()

```
async Task<string[]> ReadingBusesAPI.BusServices.BusService.GetLocationsActo ( )
```

Gets an array of stops the bus service travels too as an array of ActoCode

#### Returns

An array of Acto-Codes for the stops visited by this services.

### 7.3.3.10 GetStops()

```
async Task<List<string> > ReadingBusesAPI.BusServices.BusService.GetStops ( ) [private]
```

Gets a list of bus stops acto codes, if this is the first time it's asked for call upon the API This is delayed so only to call the API when needed.

#### Exceptions

<i>ReadingBusesApiException</i>	Thrown if you have an invalid or expired API key.
---------------------------------	---

### 7.3.3.11 GetTimeTable() [1/2]

```
Task<BusTimeTable[]> ReadingBusesAPI.BusServices.BusService.GetTimeTable (
    DateTime date )
```

Gets the full bus time table, for a specific date.

#### Parameters

<i>date</i>	the date on which you want a timetable for.
-------------	---

**Returns**

An array for the time table at a particular bus stop.

**Exceptions**

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have not provided any date.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

**7.3.3.12 GetTimeTable() [2/2]**

```
Task<BusTimeTable[ ]> ReadingBusesAPI.BusServices.BusService.GetTimeTable (
    DateTime date,
    BusStop location )
```

Gets the full bus time table, for a specific date.

**Parameters**

<i>date</i>	the date on which you want a timetable for.
<i>location</i>	(optional) a specific bus stop you want timetables for, if null it will get a timetable for every bus stop on route.

**Returns**

An array for the time table at a particular bus stop.

**Exceptions**

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have not provided any date.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

**7.3.4 Member Data Documentation****7.3.4.1 \_stops**

```
List<string> ReadingBusesAPI.BusServices.BusService._stops [private]
```

Stores a list of bus stops acto-code/IDs the service visits.



### 7.3.4.2 \_stopsObjects

```
BusStop [] ReadingBusesAPI.BusServices.BusService._stopsObjects [private]
```

Stores a list of 'BusStops' objects the service visits.

## 7.3.5 Property Documentation

### 7.3.5.1 BrandName

```
string ReadingBusesAPI.BusServices.BusService.BrandName [get], [set]
```

The brand name for the service, used mainly for Reading Buses services, such as Lion, Purple or Orange.

### 7.3.5.2 OperatorCode

```
Company ReadingBusesAPI.BusServices.BusService.OperatorCode [get], [set]
```

The operator enum value.

### 7.3.5.3 ServiceId

```
string ReadingBusesAPI.BusServices.BusService.ServiceId [get], [set]
```

The service number for the bus service, this is only guaranteed to be unique per operator, not in the API as a whole. For example Reading Buses and Newbury And District both operate a number '2' service.

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

- BusServices/[BusService.cs](#)

## 7.4 ReadingBusesAPI.BusStops.BusStop Class Reference

Stores information about a single bus stop. Related to the "List Of Bus Stops" API.

## Public Member Functions

- [BusStop](#) (string actoCode)  
*Used to create a snub/ fake object for passing to function calls, if all you need to pass is an acto-code to the function.*
- async Task< [LiveRecord](#)[]> [GetLiveData](#) ()  
*Gets live data from a bus stop.*
- [BusService](#)[] [GetServices](#) (Company busOperator)  
*Finds the 'BusService' object for all of the bus services which visit this stop.*
- [Point](#) [GetPoint](#) ()  
*Gets the geographical position of the bus stop.*
- Task< [BusTimeTable](#)[]> [GetTimeTable](#) (DateTime date)  
*Gets time table data at this specific bus stop.*
- Task< [BusTimeTable](#)[]> [GetTimeTable](#) (DateTime date, [BusService](#) service)  
*Gets time table data at this specific bus stop.*
- Task< [ArchivedBusTimeTable](#)[]> [GetArchivedTimeTable](#) (DateTime date)  
*Gets the archived real bus departure and arrival times along with their time table history at this specific bus stop.*
- Task< [ArchivedBusTimeTable](#)[]> [GetArchivedTimeTable](#) (DateTime date, [BusService](#) service)  
*Gets the archived real bus departure and arrival times along with their time table history at this specific bus stop.*

## Package Functions

- [BusStop](#) ()  
*The default constructor used for parsing data automatically.*

## Properties

- string [ActoCode](#) [get, set]  
*The unique identifier for a bus stop.*
- string [CommonName](#) [get, set]  
*The public, easy to understand stop name.*
- string [Latitude](#) [get, set]  
*The latitude of the bus stop*
- string [Longitude](#) [get, set]  
*The longitude of the bus stop*
- string [Bearing](#) [get, set]  
*The bearing of the bus stop*
- string [ServicesString](#) [get, set]
- string [GroupName](#) [get, set]  
*The Brand/Group of buses that most frequently visit this stop. Such as Purple, for the Purple 17s.*

### 7.4.1 Detailed Description

Stores information about a single bus stop. Related to the "List Of Bus Stops" API.

### 7.4.2 Constructor & Destructor Documentation

### 7.4.2.1 BusStop() [1/2]

```
ReadingBusesAPI.BusStops.BusStop.BusStop ( ) [package]
```

The default constructor used for parsing data automatically.

### 7.4.2.2 BusStop() [2/2]

```
ReadingBusesAPI.BusStops.BusStop.BusStop (
    string actCode )
```

Used to create a snub/ fake object for passing to function calls, if all you need to pass is an act-code to the function.

#### Parameters

<i>actCode</i>	ID of the bus stop.
----------------	---------------------

Unless you are doing something very strange, you probably should not need to use this, it is more for testing purposes.

## 7.4.3 Member Function Documentation

### 7.4.3.1 GetArchivedTimeTable() [1/2]

```
Task<ArchivedBusTimeTable[ ]> ReadingBusesAPI.BusStops.BusStop.GetArchivedTimeTable (
    DateTime date )
```

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

#### Parameters

<i>date</i>	The date you want time table data for. This should be a date in the past.
-------------	---

#### Returns

### 7.4.3.2 GetArchivedTimeTable() [2/2]

```
Task<ArchivedBusTimeTable[ ]> ReadingBusesAPI.BusStops.BusStop.GetArchivedTimeTable (
    DateTime date,
    BusService service )
```

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

## Parameters

<i>date</i>	The date you want time table data for. This should be a date in the past.
<i>service</i>	(optional) the service you want time table data for specifically. If null, you get time table data for all services at this stop.

## Returns

## 7.4.3.3 GetLiveData()

```
async Task<LiveRecord[]> ReadingBusesAPI.BusStops.BusStop.GetLiveData ( )
```

Gets live data from a bus stop.

## Returns

Returns a list of Live Records, which are individual buses due to arrive at the bus stop.

## 7.4.3.4 GetPoint()

```
Point ReadingBusesAPI.BusStops.BusStop.GetPoint ( )
```

Gets the geographical position of the bus stop.

## Returns

A Point Object for the position of the bus stop.

## 7.4.3.5 GetServices()

```
BusService [] ReadingBusesAPI.BusStops.BusStop.GetServices (
    Company busOperator )
```

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

## Parameters

<i>busOperator</i>	
--------------------	--

**Returns**

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

**7.4.3.6 GetTimeTable() [1/2]**

```
Task<BusTimeTable[ ]> ReadingBusesAPI.BusStops.BusStop.GetTimeTable (
    DateTime date )
```

Gets time table data at this specific bus stop.

**Parameters**

<i>date</i>	The date you want time table data for.
-------------	--

**Returns**

An array of time table records for a particular bus stop.

**Exceptions**

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have not provided any date.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

**7.4.3.7 GetTimeTable() [2/2]**

```
Task<BusTimeTable[ ]> ReadingBusesAPI.BusStops.BusStop.GetTimeTable (
    DateTime date,
    BusService service )
```

Gets time table data at this specific bus stop.

**Parameters**

<i>date</i>	The date you want time table data for.
<i>service</i>	(optional) the service you want time table data for specifically. If null, you get time table data for all services at this stop.

**Returns**

An array of time table records for a particular bus stop.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have not provided any date.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

## 7.4.4 Property Documentation

### 7.4.4.1 ActoCode

```
string ReadingBusesAPI.BusStops.BusStop.ActoCode [get], [set]
```

The unique identifier for a bus stop.

### 7.4.4.2 Bearing

```
string ReadingBusesAPI.BusStops.BusStop.Bearing [get], [set]
```

The bearing of the bus stop

### 7.4.4.3 CommonName

```
string ReadingBusesAPI.BusStops.BusStop.CommonName [get], [set]
```

The public, easy to understand stop name.

### 7.4.4.4 GroupName

```
string ReadingBusesAPI.BusStops.BusStop.GroupName [get], [set]
```

The Brand/Group of buses that most frequently visit this stop. Such as Purple, for the Purple 17s.

### 7.4.4.5 Latitude

```
string ReadingBusesAPI.BusStops.BusStop.Latitude [get], [set]
```

The latitude of the bus stop

#### 7.4.4.6 Longitude

```
string ReadingBusesAPI.BusStops.BusStop.Longitude [get], [set]
```

The longitude of the bus stop

#### 7.4.4.7 ServicesString

```
string ReadingBusesAPI.BusStops.BusStop.ServicesString [get], [set]
```

The services that travel to this stop, separated by '/'

See `BusStop.GetServices(Operators)` to get a list of Service Objects.

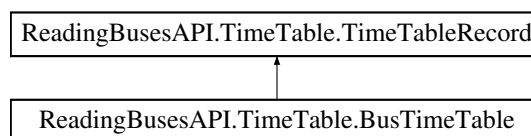
The documentation for this class was generated from the following file:

- [BusStops/BusStop.cs](#)

## 7.5 ReadingBusesAPI.TimeTable.BusTimeTable Class Reference

Represents and retrieves information about a scheduled/predicted single time table record, which means information on one bus at one location. Related to the "Timetabled Journeys" API.

Inheritance diagram for ReadingBusesAPI.TimeTable.BusTimeTable:



### Package Functions

- [BusTimeTable](#) ()  
*Default constructor to prevent creating an object directly outside the API.*

### Static Package Functions

- static async Task< [BusTimeTable](#)[]> [GetTimeTable](#) ([BusService](#) service, DateTime date, [BusStop](#) location)  
*Gets the time table of a service or a location as one array of 'BusTimeTable' objects.*
- static async Task< IGrouping< string, [BusTimeTable](#) >[]> [GetGroupedTimeTable](#) ([BusService](#) service, DateTime date, [BusStop](#) location)  
*Gets the time table for a service and groups it by a journey code instead of one continuous array of time table entries.*



## Additional Inherited Members

### 7.5.1 Detailed Description

Represents and retrieves information about a scheduled/predicted single time table record, which means information on one bus at one location. Related to the "Timetabled Journeys" API.

### 7.5.2 Constructor & Destructor Documentation

#### 7.5.2.1 BusTimeTable()

```
ReadingBusesAPI.TimeTable.BusTimeTable.BusTimeTable ( ) [package]
```

Default constructor to prevent creating an object directly outside the API.

### 7.5.3 Member Function Documentation

#### 7.5.3.1 GetGroupedTimeTable()

```
static async Task<IGrouping<string, BusTimeTable>[]> ReadingBusesAPI.TimeTable.BusTimeTable.GetGroupedTimeTable (
    BusService service,
    DateTime date,
    BusStop location ) [static], [package]
```

Gets the time table for a service and groups it by a journey code instead of one continuous array of time table entries.

#### Parameters

<i>service</i>	The bus services you wish to view.
<i>date</i>	The date of the time table.
<i>location</i>	The location to get timetable data from.

#### Returns

Returns an IGroupings of Arrays of 'BusTimeTable' records grouped by journey codes.

#### Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have not provided any date, and/or you have not provided at least either the service or location.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

### 7.5.3.2 GetTimeTable()

```
static async Task<BusTimeTable[]> ReadingBusesAPI.TimeTable.BusTimeTable.GetTimeTable (
    BusService service,
    DateTime date,
    BusStop location ) [static], [package]
```

Gets the time table of a service or a location as one array of 'BusTimeTable' objects.

#### Parameters

<i>service</i>	The bus services you wish to view.
<i>date</i>	The date of the time table.
<i>location</i>	The location to get timetable data from.

#### Returns

An array of time table records for the service or location or both

#### Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have not provided any date, and/or you have not provided at least either the service or location.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

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

- TimeTable/[BusTimeTable.cs](#)

## 7.6 ReadingBusesAPI.ErrorManagement.ErrorFormat Class Reference

Represents an error message object returned by all JSON feeds of the API.

### Properties

- bool [Status](#) [get, set]  
*The status of the request, always false for failed.*
- long? [Code](#) [get, set]  
*The status code of the error*
- string [Message](#) [get, set]  
*The error message/ reason.*

### 7.6.1 Detailed Description

Represents an error message object returned by all JSON feeds of the API.

### 7.6.2 Property Documentation

#### 7.6.2.1 Code

```
long? ReadingBusesAPI.ErrorManagement.ErrorFormat.Code [get], [set]
```

The status code of the error

#### 7.6.2.2 Message

```
string ReadingBusesAPI.ErrorManagement.ErrorFormat.Message [get], [set]
```

The error message/ reason.

#### 7.6.2.3 Status

```
bool ReadingBusesAPI.ErrorManagement.ErrorFormat.Status [get], [set]
```

The status of the request, always false for failed.

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

- ErrorManagement/[ErrorFormat.cs](#)

## 7.7 ReadingBusesAPI.ErrorManagement.ErrorManager Class Reference

Responsible for extracting and producing an error message from an API result. To the end user.

### Static Package Functions

- static void [TryErrorMessageRetrieval](#) (string json)

*Attempts to extract the error message directly sent from the API. If it can get the error message given send to the user. If the error message can not be extracted then the API has proved no explanation and so throw a generic error.*

### 7.7.1 Detailed Description

Responsible for extracting and producing an error message from an API result. To the end user.

### 7.7.2 Member Function Documentation

#### 7.7.2.1 TryErrorMessageRetrieval()

```
static void ReadingBusesAPI.ErrorManagement.ErrorManager.TryErrorMessageRetrieval (
    string json ) [static], [package]
```

Attempts to extract the error message directly sent from the API. If it can get the error message given send to the user. If the error message can not be extracted then the API has proved no explanation and so throw a generic error.

##### Parameters

<i>json</i>	Data returned from the API.
-------------	-----------------------------

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

- ErrorManagement/[ErrorManager.cs](#)

## 7.8 ReadingBusesAPI.VehiclePositions.GpsController Class Reference

Helps get live and historical GPS data on vehicles by accessing the "Live Vehicle Positions" API.

### Public Member Functions

- async Task< [ArchivedPositions\[\]](#)> [GetArchivedVehiclePositions](#) (DateTime dateStartTime, TimeSpan? timeSpan)  
*Gets historic/archived GPS data for buses on a specific date, filtered either by vehicle ID, or all buses without a time period or both. GPS data is not stored for as long as other forms of data you may fail to get data older than a few months.*
- async Task< [ArchivedPositions\[\]](#)> [GetArchivedVehiclePositions](#) (DateTime dateStartTime, TimeSpan? timeSpan, string vehicle)  
*Gets historic/archived GPS data for buses on a specific date, filtered either by vehicle ID, or all buses without a time period or both. GPS data is not stored for as long as other forms of data you may fail to get data older than a few months.*
- async Task< [LivePosition\[\]](#)> [GetLiveVehiclePositions](#) ()  
*Gets live GPS data for all buses currently operating.*
- async Task< [LivePosition](#)> [GetLiveVehiclePosition](#) (string vehicle)  
*Gets live GPS data for a single buses matching Vehicle ID number.*
- async Task< bool > [IsVehicle](#) (string vehicle)  
*Checks if the Vehicle ID Number is currently in service right now.*

## Package Functions

- [GpsController](#) ()

*Creates a GPS Controller, you should not need to make your own GPS controller, you can get an instance of one via the main '[ReadingBuses](#)' object.*

## Static Private Member Functions

- static bool [IsCacheValid](#) ()

*GPS data only updates every 30 seconds, so on average you will need to wait 15s for new data. This is used to check how long it was since last requesting GPS data. If it was recently there is no point making another request to the API as you will get the same data and take longer.*

## Private Attributes

- [LivePosition](#)[] [\\_livePositionCache](#)

*Holds the cache data for live GPS of vehicles.*

## Static Private Attributes

- static DateTime [\\_lastRetrieval](#)

*The last time a GPS request was made. This is used to prevent unnecessary API calls.*

### 7.8.1 Detailed Description

Helps get live and historical GPS data on vehicles by accessing the "Live Vehicle Positions" API.

### 7.8.2 Constructor & Destructor Documentation

#### 7.8.2.1 GpsController()

```
ReadingBusesAPI.VehiclePositions.GpsController.GpsController ( ) [package]
```

Creates a GPS Controller, you should not need to make your own GPS controller, you can get an instance of one via the main '[ReadingBuses](#)' object.

### 7.8.3 Member Function Documentation

#### 7.8.3.1 GetArchivedVehiclePositions() [1/2]

```
async Task<ArchivedPositions[]> ReadingBusesAPI.VehiclePositions.GpsController.GetArchived↵
VehiclePositions (
    DateTime dateStartTime,
    TimeSpan? timeSpan )
```

Gets historic/archived GPS data for buses on a specific date, filtered either by vehicle ID, or all buses without a time period or both. GPS data is not stored for as long as other forms of data you may fail to get data older than a few months.

## Parameters

<i>dateStartTime</i>	Vehicle ID Number eg 414
<i>timeSpan</i>	(optional) How long a period do you want data for, you can not get multiple days worth of data. If you ask this your result will be automatically truncated to only the start date to midnight.

## Returns

An array of GPS locations at a previous date.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if, you have not choose a date in the past, or the date is too far in the past and so no data exists. Thrown if you have not filtered by either 'timeSpan' or 'vehicle' ID or both. Thrown if the API key is invalid or expired.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

See [GpsController.GetLiveVehiclePositions\(\)](#) to get live data instead.

**7.8.3.2 GetArchivedVehiclePositions()** [2/2]

```
async Task<ArchivedPositions[]> ReadingBusesAPI.VehiclePositions.GpsController.GetArchived↵
VehiclePositions (
    DateTime dateStartTime,
    TimeSpan? timeSpan,
    string vehicle )
```

Gets historic/archived GPS data for buses on a specific date, filtered either by vehicle ID, or all buses without a time period or both. GPS data is not stored for as long as other forms of data you may fail to get data older than a few months.

## Parameters

<i>dateStartTime</i>	Vehicle ID Number eg 414
<i>timeSpan</i>	(optional) How long a period do you want data for, you can not get multiple days worth of data. If you ask this your result will be automatically truncated to only the start date to midnight.
<i>vehicle</i>	(optional) Vehicle ID Number eg 414

## Returns

An array of GPS locations at a previous date.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if, you have not choose a date in the past, or the date is too far in the past and so no data exists. Thrown if you have not filtered by either 'timeSpan' or 'vehicle' ID or both. Thrown if the API key is invalid or expired.
---	---

## Exceptions

<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

See [GpsController.GetLiveVehiclePositions\(\)](#) to get live data instead.

**7.8.3.3 GetLiveVehiclePosition()**

```
async Task<LivePosition> ReadingBusesAPI.VehiclePositions.GpsController.GetLiveVehiclePosition
(
    string vehicle )
```

Gets live GPS data for a single buses matching Vehicle ID number.

## Parameters

<i>vehicle</i>	Vehicle ID Number eg 414
----------------	--------------------------

## Returns

The GPS point of Vehicle matching your ID provided.

## Exceptions

<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if a vehicle of the ID does not exist or is not currently active. You can check by using the 'IsVehicle' function.
---	---

**7.8.3.4 GetLiveVehiclePositions()**

```
async Task<LivePosition[]> ReadingBusesAPI.VehiclePositions.GpsController.GetLiveVehicle↵
Positions ( )
```

Gets live GPS data for all buses currently operating.

## Returns

An array of GPS locations for all buses operating by Reading Buses currently

## Exceptions

<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API key is invalid or expired.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

### 7.8.3.5 IsCacheValid()

```
static bool ReadingBusesAPI.VehiclePositions.GpsController.IsCacheValid ( ) [static], [private]
```

GPS data only updates every 30 seconds, so on average you will need to wait 15s for new data. This is used to check how long it was since last requesting GPS data. If it was recently there is no point making another request to the API as you will get the same data and take longer.

#### Returns

Returns if it has been less than 15 seconds from last asking for GPS data.

### 7.8.3.6 IsVehicle()

```
async Task<bool> ReadingBusesAPI.VehiclePositions.GpsController.IsVehicle (
    string vehicle )
```

Checks if the Vehicle ID Number is currently in service right now.

#### Parameters

<i>vehicle</i>	Vehicle ID Number eg 414
----------------	--------------------------

#### Returns

True or False for if the buses GPS can be found or not currently.

## 7.8.4 Member Data Documentation

### 7.8.4.1 \_lastRetrieval

```
DateTime ReadingBusesAPI.VehiclePositions.GpsController._lastRetrieval [static], [private]
```

The last time a GPS request was made. This is used to prevent unnecessary API calls.

### 7.8.4.2 \_livePositionCache

```
LivePosition [ ] ReadingBusesAPI.VehiclePositions.GpsController._livePositionCache [private]
```

Holds the cache data for live GPS of vehicles.

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

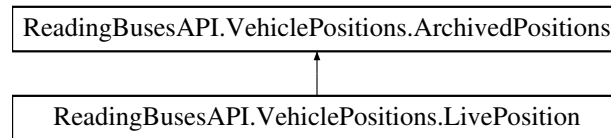
- VehiclePositions/[GPSController.cs](#)



## 7.9 ReadingBusesAPI.VehiclePositions.LivePosition Class Reference

Used to store live information about a buses GPS position. Related to the "Live Vehicle Positions" API.

Inheritance diagram for ReadingBusesAPI.VehiclePositions.LivePosition:



### Public Member Functions

- [BusService GetService](#) ()  
*Finds the 'BusService' object related to the record.*

### Package Functions

- [LivePosition](#) ()  
*The default constructor, which sets the 'LastRetrieval' to current time.*

### Properties

- string [ServiceId](#) [get, set]  
*Holds the Service Number for the bus route.*
- string [Bearing](#) [get, set]  
*bearing direction of the bus*

#### 7.9.1 Detailed Description

Used to store live information about a buses GPS position. Related to the "Live Vehicle Positions" API.

#### 7.9.2 Constructor & Destructor Documentation

##### 7.9.2.1 LivePosition()

```
ReadingBusesAPI.VehiclePositions.LivePosition.LivePosition ( ) [package]
```

The default constructor, which sets the 'LastRetrieval' to current time.

### 7.9.3 Member Function Documentation

#### 7.9.3.1 GetService()

```
BusService ReadingBusesAPI.VehiclePositions.LivePosition.GetService ( )
```

Finds the 'BusService' object related to the record.

##### Returns

The related 'BusService' object.

### 7.9.4 Property Documentation

#### 7.9.4.1 Bearing

```
string ReadingBusesAPI.VehiclePositions.LivePosition.Bearing [get], [set]
```

bearing direction of the bus

#### 7.9.4.2 ServiceId

```
string ReadingBusesAPI.VehiclePositions.LivePosition.ServiceId [get], [set]
```

Holds the Service Number for the bus route.

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

- VehiclePositions/[LivePosition.cs](#)

## 7.10 ReadingBusesAPI.JourneyDetails.LiveRecord Class Reference

Used to store information about a buses arrival at a bus stop. Mainly related to the "Stop Predictions" API.

### Public Member Functions

- [BusService Service](#) ()  
*Returns the related BusService Object for the Bus [LiveRecord](#).*
- string [DisplayTime](#) ()  
*Returns the number of min till bus is due in a min format.*
- double [ArrivalMin](#) ()  
*Returns the number of min till the bus is due to arrive.*

## Package Functions

- [LiveRecord](#) ()

*The default constructor, used for XML parsing.*

## Static Package Functions

- static [LiveRecord](#)[] [GetLiveData](#) (string actCode)

*Gets a list of upcoming arrivals at a specific bus stop. Can throw an exception.*

## Properties

- string [ServiceNumber](#) [get, set]  
*Holds the Service Number for the bus route.*
- string [Destination](#) [get, set]  
*Holds the destination for the bus.*
- DateTime [SchArrival](#) [get, set]  
*Holds scheduled arrival time of the bus at the location.*
- DateTime? [ExptArrival](#) [get, set]  
*Holds the estimated/ expected arrival time of the bus, if Null no estimated time exists yet.*
- [Company OperatorCode](#) [get, set]  
*Holds the operator of the service.*
- string [VehicleRef](#) [get, set]  
*Holds the Vehicles reference ID or number to identify it.*
- string [ViaMessage](#) [get, set]

### 7.10.1 Detailed Description

Used to store information about a buses arrival at a bus stop. Mainly related to the "Stop Predictions" API.

### 7.10.2 Constructor & Destructor Documentation

#### 7.10.2.1 LiveRecord()

```
ReadingBusesAPI.JourneyDetails.LiveRecord.LiveRecord ( ) [package]
```

The default constructor, used for XML parsing.

### 7.10.3 Member Function Documentation

### 7.10.3.1 ArrivalMin()

```
double ReadingBusesAPI.JourneyDetails.LiveRecord.ArrivalMin ( )
```

Returns the number of min till the bus is due to arrive.

#### Returns

The number of min till the bus is due to arrive.

### 7.10.3.2 DisplayTime()

```
string ReadingBusesAPI.JourneyDetails.LiveRecord.DisplayTime ( )
```

Returns the number of min till bus is due in a min format.

#### Returns

The number of min until the bus is due to arrive in string format.

### 7.10.3.3 GetLiveData()

```
static LiveRecord [] ReadingBusesAPI.JourneyDetails.LiveRecord.GetLiveData (
    string actoCode ) [static], [package]
```

Gets a list of upcoming arrivals at a specific bus stop. Can throw an exception.

#### Parameters

<i>actoCode</i>	The Acto-code ID for a specific bus stop.
-----------------	---

#### Returns

A list of Live Records containing details about upcoming buses.

#### Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if no data is returned from the API.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if you have used an invalid or expired API key or an invalid acto-code
<i>ReadingBusesApiExceptionCritical</i>	Thrown if no error message or reasoning for fault is detectable.

### 7.10.3.4 Service()

`BusService` ReadingBusesAPI.JourneyDetails.LiveRecord.Service ( )

Returns the related BusService Object for the Bus [LiveRecord](#).

#### Returns

Information about the current bus service object.

#### Exceptions

<i>InvalidOperationException</i>	Can throw an exception if the service does not exists. This is however very unlikely, if this occurs there is an error in the API, not with your code.
----------------------------------	--

## 7.10.4 Property Documentation

### 7.10.4.1 Destination

`string` ReadingBusesAPI.JourneyDetails.LiveRecord.Destination [get], [set]

Holds the destination for the bus.

### 7.10.4.2 ExptArrival

`DateTime?` ReadingBusesAPI.JourneyDetails.LiveRecord.ExptArrival [get], [set]

Holds the estimated/ expected arrival time of the bus, if Null no estimated time exists yet.

### 7.10.4.3 OperatorCode

`Company` ReadingBusesAPI.JourneyDetails.LiveRecord.OperatorCode [get], [set]

Holds the operator of the service.

### 7.10.4.4 SchArrival

`DateTime` ReadingBusesAPI.JourneyDetails.LiveRecord.SchArrival [get], [set]

Holds scheduled arrival time of the bus at the location.

#### 7.10.4.5 ServiceNumber

```
string ReadingBusesAPI.JourneyDetails.LiveRecord.ServiceNumber [get], [set]
```

Holds the Service Number for the bus route.

#### 7.10.4.6 VehicleRef

```
string ReadingBusesAPI.JourneyDetails.LiveRecord.VehicleRef [get], [set]
```

Holds the Vehicles reference ID or number to identify it.

#### 7.10.4.7 ViaMessage

```
string ReadingBusesAPI.JourneyDetails.LiveRecord.ViaMessage [get], [set]
```

Holds the 'Via' message, which explains where the bus is traveling past on route. Can be null or a place holder value if none exists.

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

- JourneyDetails/[LiveRecord.cs](#)

## 7.11 ReadingBusesAPI.BusStops.Locations Class Reference

This classes simply gets all the buses stops visited by Reading Buses, by interfacing with the "List Of Bus Stops" API.

### Package Functions

- `async Task< Dictionary< string, BusStop > > FindLocations ()`  
*Finds all the bus stops visited by Reading Buses.*

### Static Private Attributes

- `const string CacheLocation = "cache\\Locations.cache"`  
*the location for the service cache file.*

#### 7.11.1 Detailed Description

This classes simply gets all the buses stops visited by Reading Buses, by interfacing with the "List Of Bus Stops" API.

#### 7.11.2 Member Function Documentation

##### 7.11.2.1 FindLocations()

```
async Task<Dictionary<string, BusStop> > ReadingBusesAPI.BusStops.Locations.FindLocations ( )  
[package]
```

Finds all the bus stops visited by Reading Buses.

## Exceptions

<i>InvalidOperationException</i>	Thrown if an invalid or expired API Key is used.
----------------------------------	--

### 7.11.3 Member Data Documentation

#### 7.11.3.1 CacheLocation

```
const string ReadingBusesAPI.BusStops.Locations.CacheLocation = "cache\\Locations.cache" [static],
[private]
```

the location for the service cache file.

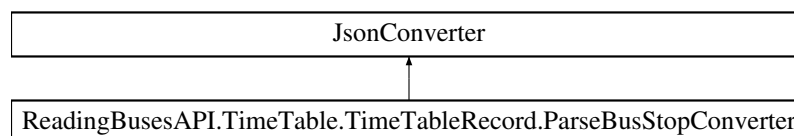
The documentation for this class was generated from the following file:

- BusStops/[Locations.cs](#)

## 7.12 ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter Class Reference

Converts a bus stop acto-code into a 'BusStop' Object and back again for the JSON converter.

Inheritance diagram for ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter:



### Public Member Functions

- override bool [CanConvert](#) (Type t)
- override object [ReadJson](#) (JsonReader reader, Type t, object existingValue, JsonSerializer serializer)
- override void [WriteJson](#) (JsonWriter writer, object untypedValue, JsonSerializer serializer)

### Static Public Attributes

- static readonly [ParseBusStopConverter Singleton](#) = new [ParseBusStopConverter](#)()

### 7.12.1 Detailed Description

Converts a bus stop acto-code into a 'BusStop' Object and back again for the JSON converter.

### 7.12.2 Member Function Documentation

#### 7.12.2.1 CanConvert()

```
override bool ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter.CanConvert (
    Type t )
```

#### 7.12.2.2 ReadJson()

```
override object ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter.ReadJson (
    JsonReader reader,
    Type t,
    object existingValue,
    JsonSerializer serializer )
```

#### 7.12.2.3 WriteJson()

```
override void ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter.WriteJson (
    JsonWriter writer,
    object untypedValue,
    JsonSerializer serializer )
```

### 7.12.3 Member Data Documentation

#### 7.12.3.1 Singleton

```
readonly ParseBusStopConverter ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter.↵
Singleton = new ParseBusStopConverter() [static]
```

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

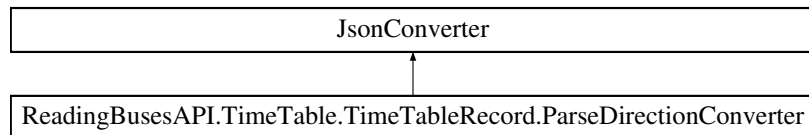
- [TimeTable/TimeTableRecord.cs](#)



## 7.13 ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter Class Reference

Converts a string into a Direction Enum and back again for the JSON converter.

Inheritance diagram for ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter:



### Public Member Functions

- override bool [CanConvert](#) (Type t)
- override object [ReadJson](#) (JsonReader reader, Type t, object existingValue, JsonSerializer serializer)
- override void [WriteJson](#) (JsonWriter writer, object untypedValue, JsonSerializer serializer)

### Static Public Attributes

- static readonly [ParseDirectionConverter Singleton](#) = new [ParseDirectionConverter](#)()

#### 7.13.1 Detailed Description

Converts a string into a Direction Enum and back again for the JSON converter.

#### 7.13.2 Member Function Documentation

##### 7.13.2.1 CanConvert()

```
override bool ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter.CanConvert (
    Type t )
```

##### 7.13.2.2 ReadJson()

```
override object ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter.ReadJson (
    JsonReader reader,
    Type t,
    object existingValue,
    JsonSerializer serializer )
```

### 7.13.2.3 WriteJson()

```
override void ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter.WriteJson (
    JsonWriter writer,
    object untypedValue,
    JsonSerializer serializer )
```

## 7.13.3 Member Data Documentation

### 7.13.3.1 Singleton

```
readonly ParseDirectionConverter ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirection↔
Converter.Singleton = new ParseDirectionConverter() [static]
```

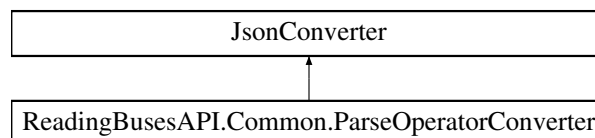
The documentation for this class was generated from the following file:

- TimeTable/[TimeTableRecord.cs](#)

## 7.14 ReadingBusesAPI.Common.ParseOperatorConverter Class Reference

Converts a string short code for an Operator into an Operator Enum and back again for the JSON converter.

Inheritance diagram for ReadingBusesAPI.Common.ParseOperatorConverter:



### Public Member Functions

- override bool [CanConvert](#) (Type t)
- override object [ReadJson](#) (JsonReader reader, Type t, object existingValue, JsonSerializer serializer)
- override void [WriteJson](#) (JsonWriter writer, object untypedValue, JsonSerializer serializer)

### Static Public Attributes

- static readonly [ParseOperatorConverter Singleton](#) = new [ParseOperatorConverter](#)()

### 7.14.1 Detailed Description

Converts a string short code for an Operator into an Operator Enum and back again for the JSON converter.

## 7.14.2 Member Function Documentation

### 7.14.2.1 CanConvert()

```
override bool ReadingBusesAPI.Common.ParseOperatorConverter.CanConvert (
    Type t )
```

### 7.14.2.2 ReadJson()

```
override object ReadingBusesAPI.Common.ParseOperatorConverter.ReadJson (
    JsonReader reader,
    Type t,
    object existingValue,
    JsonSerializer serializer )
```

### 7.14.2.3 WriteJson()

```
override void ReadingBusesAPI.Common.ParseOperatorConverter.WriteJson (
    JsonWriter writer,
    object untypedValue,
    JsonSerializer serializer )
```

## 7.14.3 Member Data Documentation

### 7.14.3.1 Singleton

```
readonly ParseOperatorConverter ReadingBusesAPI.Common.ParseOperatorConverter.Singleton = new
ParseOperatorConverter() [static]
```

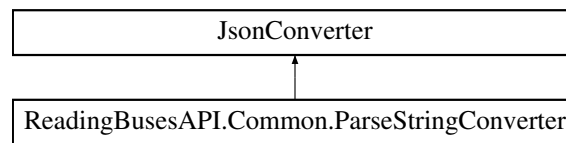
The documentation for this class was generated from the following file:

- Common/[JSONConverters.cs](#)

## 7.15 ReadingBusesAPI.Common.ParseStringConverter Class Reference

Converts a string into a long and back again for the JSON converter.

Inheritance diagram for ReadingBusesAPI.Common.ParseStringConverter:



### Public Member Functions

- override bool [CanConvert](#) (Type t)
- override object [ReadJson](#) (JsonReader reader, Type t, object existingValue, JsonSerializer serializer)
- override void [WriteJson](#) (JsonWriter writer, object untypedValue, JsonSerializer serializer)

### Static Public Attributes

- static readonly [ParseStringConverter Singleton](#) = new [ParseStringConverter](#)()

#### 7.15.1 Detailed Description

Converts a string into a long and back again for the JSON converter.

#### 7.15.2 Member Function Documentation

##### 7.15.2.1 CanConvert()

```

override bool ReadingBusesAPI.Common.ParseStringConverter.CanConvert (
    Type t )
  
```

##### 7.15.2.2 ReadJson()

```

override object ReadingBusesAPI.Common.ParseStringConverter.ReadJson (
    JsonReader reader,
    Type t,
    object existingValue,
    JsonSerializer serializer )
  
```

### 7.15.2.3 WriteJson()

```
override void ReadingBusesAPI.Common.ParseStringConverter.WriteJson (
    JsonWriter writer,
    object untypedValue,
    JsonSerializer serializer )
```

## 7.15.3 Member Data Documentation

### 7.15.3.1 Singleton

```
readonly ParseStringConverter ReadingBusesAPI.Common.ParseStringConverter.Singleton = new
ParseStringConverter() [static]
```

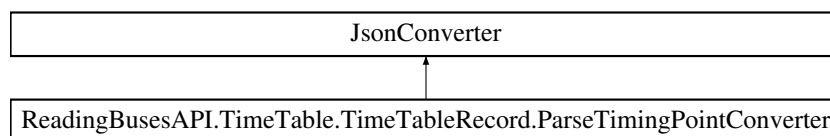
The documentation for this class was generated from the following file:

- Common/[JSONConverters.cs](#)

## 7.16 ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter Class Reference

Converts a string into a boolean and back again for the JSON converter.

Inheritance diagram for ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter:



### Public Member Functions

- override bool [CanConvert](#) (Type t)
- override object [ReadJson](#) (JsonReader reader, Type t, object existingValue, JsonSerializer serializer)
- override void [WriteJson](#) (JsonWriter writer, object untypedValue, JsonSerializer serializer)

### Static Public Attributes

- static readonly [ParseTimingPointConverter Singleton](#) = new [ParseTimingPointConverter](#)()

### 7.16.1 Detailed Description

Converts a string into a boolean and back again for the JSON converter.

### 7.16.2 Member Function Documentation

#### 7.16.2.1 CanConvert()

```
override bool ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter.CanConvert (
    Type t )
```

#### 7.16.2.2 ReadJson()

```
override object ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter.ReadJson (
    JsonReader reader,
    Type t,
    object existingValue,
    JsonSerializer serializer )
```

#### 7.16.2.3 WriteJson()

```
override void ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter.WriteJson (
    JsonWriter writer,
    object untypedValue,
    JsonSerializer serializer )
```

### 7.16.3 Member Data Documentation

#### 7.16.3.1 Singleton

```
readonly ParseTimingPointConverter ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter.Singleton = new ParseTimingPointConverter() [static]
```

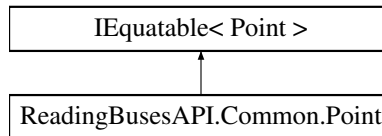
The documentation for this class was generated from the following file:

- [TimeTable/TimeTableRecord.cs](#)

## 7.17 ReadingBusesAPI.Common.Point Struct Reference

Stores an X and Y Position simply.

Inheritance diagram for ReadingBusesAPI.Common.Point:



### Public Member Functions

- [Point](#) (double x, double y)  
*Default constructor*
- override string [ToString](#) ()  
*Converts point to string representation.*
- override int [GetHashCode](#) ()  
*Generates a unique number for each [Point](#) Object.*
- override bool [Equals](#) (object obj)  
*Checks if two point objects are the same or not.*
- bool [Equals](#) ([Point](#) other)  
*Implements logic for checking if two objects are equal.*

### Static Public Member Functions

- static bool [operator==](#) ([Point](#) point1, [Point](#) point2)  
*Checks if two objects are the same.*
- static bool [operator!=](#) ([Point](#) point1, [Point](#) point2)  
*Checks if two objects are the not the same.*

### Properties

- double [X](#) [get]  
*The X value.*
- double [Y](#) [get]  
*The Y Value.*

#### 7.17.1 Detailed Description

Stores an X and Y Position simply.

#### 7.17.2 Constructor & Destructor Documentation

##### 7.17.2.1 Point()

```

ReadingBusesAPI.Common.Point.Point (
    double x,
    double y )
  
```

Default constructor

**Parameters**

<i>x</i>	X value of <a href="#">Point</a> .
<i>y</i>	Y value of <a href="#">Point</a> .

### 7.17.3 Member Function Documentation

#### 7.17.3.1 Equals() [1/2]

```
override bool ReadingBusesAPI.Common.Point.Equals (
    object obj )
```

Checks if two point objects are the same or not.

**Parameters**

<i>obj</i>	Another object to compare against this object.
------------	--

**Returns**

Is it the same object or not.

#### 7.17.3.2 Equals() [2/2]

```
bool ReadingBusesAPI.Common.Point.Equals (
    Point other )
```

Implements logic for checking if two objects are equal.

**Parameters**

<i>other</i>	The other object to check if equal.
--------------	-------------------------------------

**Returns**

True if equals else false.

#### 7.17.3.3 GetHashCode()

```
override int ReadingBusesAPI.Common.Point.GetHashCode ( )
```

Generates a unique number for each [Point](#) Object.



**Returns**

Int value of object.

**7.17.3.4 operator"!=()"**

```
static bool ReadingBusesAPI.Common.Point.operator!= (
    Point point1,
    Point point2 ) [static]
```

Checks if two objects are the not the same.

**Parameters**

<i>point1</i>	First <a href="#">Point</a> Object.
<i>point2</i>	Second <a href="#">Point</a> Object.

**Returns**

True if they are not the same.

**7.17.3.5 operator=="()**

```
static bool ReadingBusesAPI.Common.Point.operator== (
    Point point1,
    Point point2 ) [static]
```

Checks if two objects are the same.

**Parameters**

<i>point1</i>	First <a href="#">Point</a> Object.
<i>point2</i>	Second <a href="#">Point</a> Object.

**Returns**

True if equal else false

**7.17.3.6 ToString()**

```
override string ReadingBusesAPI.Common.Point.ToString ( )
```

Converts point to string representation.

**Returns**

[Point](#) as a string.

**7.17.4 Property Documentation****7.17.4.1 X**

```
double ReadingBusesAPI.Common.Point.X [get]
```

The X value.

**7.17.4.2 Y**

```
double ReadingBusesAPI.Common.Point.Y [get]
```

The Y Value.

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

- Common/[Point.cs](#)

**7.18 ReadingBusesAPI.ReadingBuses Class Reference**

This is the main class for the library, here you can initialise a singleton instance and then query and use the Reading Buses API.

**Public Member Functions**

- [BusStop GetLocation](#) (string actoCode)  
*Get a bus stop location based upon a bus stops location code*
- [BusStop\[\] GetLocations](#) ()  
*All the bus stop locations that Reading Buses Visits*
- bool [IsLocation](#) (string actoCode)  
*Checks to see if the acto code for the bus stop exists in the API feed or not.*
- [BusService\[\] GetServices](#) ()  
*All the Services Reading Buses Operates*
- [BusService\[\] GetServices](#) (string brandName)  
*Returns all services Reading Buses Operates under a brand name, for example "pink" would return "22,25,27,29" services.*
- [BusService\[\] GetServices](#) (Company operatorCode)  
*Returns all services Reading Buses Operates under a specific operator.*
- [BusService\[\] GetService](#) (string serviceNumber)

*Returns a service which matches the Service Number passed, because the Reading Buses API now supports, Kennections and Newbury and District a service number can no longer be considered unique.*

- [BusService.GetService](#) (string serviceNumber, [Company](#) operators)

*Returns a service which matches the Service Number passed and the bus operator.*

- bool [IsService](#) (string serviceNumber)

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

- bool [IsService](#) (string serviceNumber, [Company](#) operators)

*Checks to see if a service of that number exists or not in the API feed, for a specific bus operator.*

- Task< [ArchivedBusTimeTable\[\]](#)> [GetVehicleTrackingHistory](#) (DateTime date, string vehicle)

*Gets the archived real bus departure and arrival times along with their time table history for a specific vehicle, on a specific date. This can be used to find how late a vehicle was throughout that day.*

## Static Public Member Functions

- static void [SetCache](#) (bool value)

*Sets if you want to cache data into local files or always get new data from the API, which will take longer.*

- static void [SetDebugging](#) (bool value)

*Sets if you want to debug the library by making requests to a dummy server instead of the real live sever.*

- static void [SetWarning](#) (bool value)

*Sets if you want to print out warning messages to the console screen or not. Only done so in debug.*

- static void [SetFullError](#) (bool value)

*Sets if you want to print out the full error logs to console, only needed for debugging library errors. Only done so in debug.*

- static void [SetCacheValidityLength](#) (int days)

*Sets how long to keep Cache data for before invalidating it and getting new data.*

- static void [InvalidateCache](#) ()

*Deletes any Cache data stored, Cache data is deleted automatically after a number of days, use this only if you need to force new data early.*

- static async Task< [ReadingBuses](#) > [Initialise](#) (string apiKey)

*Used to initially initialise the [ReadingBuses](#) Object, it is recommended you do this in your programs start up.*

- static [ReadingBuses](#) [GetInstance](#) ()

*You will never need more than one [ReadingBuses](#) object, a singleton is used to ensure you always get the same instance.*

## Static Package Functions

- static void [PrintWarning](#) (string message)

*Internal method for printing warning messages to the console screen. Only done so in debug.*

- static void [PrintFullErrorLogs](#) (string message)

*Internal method for printing warning messages to the console screen. Only done so in debug.*

- static [Company](#) [GetOperatorE](#) (string operatorCodeS)

*Converts the short hand code for an operator into its Enum value, for example RGB stands for Reading Buses.*

## Properties

- static bool [Debugging](#) [get, private set]  
*Keeps track of if cache data is being used or not*
- static bool [Cache](#) = true [get, private set]  
*Keeps track of if cache data is being used or not*
- static bool [Warning](#) = true [get, private set]  
*Keeps track of if warnings are being outputted to console or not.*
- static bool [FullError](#) [get, private set]  
*Keeps track of if full error logs are being outputted to console or not.*
- static int [CacheValidityLength](#) = 7 [get, private set]  
*Stores how many days cache data is valid for in days before being regenerated*
- static string [ApiKey](#) [get, private set]  
*Holds the users API Key.*
- [GpsController GpsController](#) [get]  
*Stores the GPS controller, which can help get vehicle GPS data.*

## Private Member Functions

- [ReadingBuses](#) (string apiKey)  
*Create a new Reading Buses library object, this is the main control.*
- async Task [SetUp](#) ()  
*Creates cache data and retrieves bus services and bus stop data.*

## Static Private Member Functions

- static void [DeleteInstance](#) ()  
*Deletes the current singleton instance if there was an error generating one.*

## Private Attributes

- Dictionary< string, [BusStop](#) > [\\_locations](#)  
*Holds information on all the bus stops/locations visited by Reading Buses*
- List< [BusService](#) > [\\_services](#)  
*Holds information on all the services operated by Reading Buses*

## Static Private Attributes

- static [ReadingBuses \\_instance](#)  
*The singleton instance*

### 7.18.1 Detailed Description

This is the main class for the library, here you can initialise a singleton instance and then query and use the Reading Buses API.

```
//Optional
ReadingBuses.SetCache(true);
ReadingBuses Controller = await ReadingBuses.Initialise("API KEY HERE");
BusService service = Controller.GetService("17"); or ReadingBuses.GetInstance().GetService("17");
```

Cached Data is data stored locally in JSON and XML files, stored in a hidden folder called "cache", in the same directory the program is executed from. This is a copy of the results from an API call, such as the bus services and bus stops, because it is unlikely for this data to change regularly. By default the cached data will be updated every 7 days, but you can request new data or disable cache if you wish. Caching data is however faster as you do not need to keep making API requests for data likely to be the same.

## 7.18.2 Constructor & Destructor Documentation

### 7.18.2.1 ReadingBuses()

```
ReadingBusesAPI.ReadingBuses.ReadingBuses (
    string apiKey ) [private]
```

Create a new Reading Buses library object, this is the main control.

#### Parameters

<i>apiKey</i>	The Reading Buses API Key, get your own from <a href="http://rtl2.ods-live.co.uk/cms/apiservice">http://rtl2.ods-live.co.uk/cms/apiservice</a>
---------------	--

## 7.18.3 Member Function Documentation

### 7.18.3.1 DeleteInstance()

```
static void ReadingBusesAPI.ReadingBuses.DeleteInstance ( ) [static], [private]
```

Deletes the current singleton instance if there was an error generating one.

### 7.18.3.2 GetInstance()

```
static ReadingBuses ReadingBusesAPI.ReadingBuses.GetInstance ( ) [static]
```

You will never need more than one [ReadingBuses](#) object, a singleton is used to ensure you always get the same instance.

#### Returns

Returns the [ReadingBuses](#) object to be used throughout your program.

#### Exceptions

<i>InvalidOperationException</i>	Thrown if you attempt to get an instance before you have called the "Initialise" function.
----------------------------------	--

See [ReadingBuses.Initialise\(string\)](#) to initially initialise the [ReadingBuses](#) Object singleton.

### 7.18.3.3 GetLocation()

```
BusStop ReadingBusesAPI.ReadingBuses.GetLocation (
    string actoCode )
```

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

#### Parameters

<i>actoCode</i>	The code of the bus stop
-----------------	--------------------------

#### Returns

A Bus Stop object for the Acto Code specified.

#### Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if the bus stop does not exist. You should first check with 'IsLocation' If there is any uncertainty.
---	--

See [ReadingBuses.IsLocation\(string\)](#) to check if it is a location.

### 7.18.3.4 GetLocations()

```
BusStop [ ] ReadingBusesAPI.ReadingBuses.GetLocations ( )
```

All the bus stop locations that Reading Buses Visits

#### Returns

All the bus stops Reading Buses visits

### 7.18.3.5 GetOperatorE()

```
static Company ReadingBusesAPI.ReadingBuses.GetOperatorE (
    string operatorCodeS ) [static], [package]
```

Converts the short hand code for an operator into its Enum value, for example RGB stands for Reading Buses.

#### Parameters

<i>operatorCodeS</i>	
----------------------	--

**Returns**

The Enum equivalent of the bus operator short code.

**7.18.3.6 GetService() [1/2]**

```
BusService [ ] ReadingBusesAPI.ReadingBuses.GetService (
    string serviceName )
```

Returns a service which matches the Service Number passed, because the Reading Buses API now supports, Kennections and Newbury and District a service number can no longer be considered unique.

**Parameters**

<i>serviceName</i>	The service number/ID for the service you wish to be returned eg: 17 or 22.
--------------------	---

**Returns**

The services matching the ID.

**Exceptions**

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if the bus services does not exist. You should first check with 'IsService' If there is any uncertainty.
---	---

See [ReadingBuses.IsService\(string\)](#) to check if it is a service.

**7.18.3.7 GetService() [2/2]**

```
BusService ReadingBusesAPI.ReadingBuses.GetService (
    string serviceName,
    Company operators )
```

Returns a service which matches the Service Number passed and the bus operator.

**Parameters**

<i>serviceName</i>	The service number/ID for the service you wish to be returned eg: 17 or 22.
<i>operators</i>	The bus operator to search in, for example "ReadingBuses"

**Returns**

The services matching the ID.

## Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if the bus services does not exist. You should first check with 'IsService' If there is any uncertainty.
---	---

See [ReadingBuses.IsService\(string\)](#) to check if it is a service.

**7.18.3.8 GetServices()** [1/3]

```
BusService [ ] ReadingBusesAPI.ReadingBuses.GetServices ( )
```

All the Services Reading Buses Operates

## Returns

All the Services Reading Buses Operates

**7.18.3.9 GetServices()** [2/3]

```
BusService [ ] ReadingBusesAPI.ReadingBuses.GetServices (
    Company operatorCode )
```

Returns all services Reading Buses Operates under a specific operator.

## Parameters

<i>operatorCode</i>	The operator to filter by.
---------------------	----------------------------

## Returns

An array of Bus Services which are of the brand name specified.

**7.18.3.10 GetServices()** [3/3]

```
BusService [ ] ReadingBusesAPI.ReadingBuses.GetServices (
    string brandName )
```

Returns all services Reading Buses Operates under a brand name, for example "pink" would return "22,25,27,29" services.

## Parameters

<i>brandName</i>	The brand name for the services you wish to find, eg "pink" or "sky blue".
------------------	--



**Returns**

An array of Bus Services which are of the brand name specified.

**7.18.3.11 GetVehicleTrackingHistory()**

```
Task<ArchivedBusTimeTable[]> ReadingBusesAPI.ReadingBuses.GetVehicleTrackingHistory (
    DateTime date,
    string vehicle )
```

Gets the archived real bus departure and arrival times along with their time table history for a specific vehicle, on a specific date. This can be used to find how late a vehicle was throughout that day.

**Parameters**

<i>date</i>	The date you want a report for, must be in the past.
<i>vehicle</i>	The vehicle ID number

**Returns**

An array of Archived Bus Departure and arrival times with their timetabled data.

**Exceptions**

<i>ReadingBusesApiExceptionMalformedQuery</i>	If you have tried to get data for a date in the future. Or if you have not provided any date, and/or you have not provided at least either the service or location or vehicle.
<i>ReadingBusesApiExceptionBadQuery</i>	Thrown if the API responds with an error message.
<i>ReadingBusesApiExceptionCritical</i>	Thrown if the API fails, but provides no reason.

**7.18.3.12 Initialise()**

```
static async Task<ReadingBuses> ReadingBusesAPI.ReadingBuses.Initialise (
    string apiKey ) [static]
```

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

**Parameters**

<i>apiKey</i>	The Reading Buses API Key, get your own from <a href="http://rt12.ods-live.co.uk/cms/apiservice">http://rt12.ods-live.co.uk/cms/apiservice</a>
---------------	--

**Returns**

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

## Exceptions

<i>ReadingBusesApiExceptionBadQuery</i>	Can throw an exception if you pass an invalid or expired API Key.
---	---

See [ReadingBuses.GetInstance\(\)](#) to get any future instances afterwards.

**7.18.3.13 InvalidateCache()**

```
static void ReadingBusesAPI.ReadingBuses.InvalidateCache ( ) [static]
```

Deletes any Cache data stored, Cache data is deleted automatically after a number of days, use this only if you need to force new data early.

**7.18.3.14 IsLocation()**

```
bool ReadingBusesAPI.ReadingBuses.IsLocation (
    string actoCode )
```

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

## Parameters

<i>actoCode</i>	The ID Code for a bus stop.
-----------------	-----------------------------

## Returns

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

**7.18.3.15 IsService() [1/2]**

```
bool ReadingBusesAPI.ReadingBuses.IsService (
    string serviceName )
```

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

## Parameters

<i>serviceName</i>	The service number to find.
--------------------	-----------------------------

## Returns

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

### 7.18.3.16 IsService() [2/2]

```
bool ReadingBusesAPI.ReadingBuses.IsService (
    string serviceNumber,
    Company operators )
```

Checks to see if a service of that number exists or not in the API feed, for a specific bus operator.

#### Parameters

<i>serviceNumber</i>	The service number to find.
<i>operators</i>	The specific bus operator you want to search in.

#### Returns

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

### 7.18.3.17 PrintFullErrorLogs()

```
static void ReadingBusesAPI.ReadingBuses.PrintFullErrorLogs (
    string message ) [static], [package]
```

Internal method for printing warning messages to the console screen. Only done so in debug.

#### Parameters

<i>message</i>	The message to print off to console.
----------------	--------------------------------------

### 7.18.3.18 PrintWarning()

```
static void ReadingBusesAPI.ReadingBuses.PrintWarning (
    string message ) [static], [package]
```

Internal method for printing warning messages to the console screen. Only done so in debug.

#### Parameters

<i>message</i>	The message to print off to console.
----------------	--------------------------------------

### 7.18.3.19 SetCache()

```
static void ReadingBusesAPI.ReadingBuses.SetCache (
```

```
bool value ) [static]
```

Sets if you want to cache data into local files or always get new data from the API, which will take longer.

#### Parameters

<i>value</i>	True or False for if you want to get Cache or live data.
--------------	--

#### Exceptions

<i>ReadingBusesApiExceptionMalformedQuery</i>	Thrown if you attempt to change the cache options after the library has been instantiated
---	---

### 7.18.3.20 SetCacheValidityLength()

```
static void ReadingBusesAPI.ReadingBuses.SetCacheValidityLength (
    int days ) [static]
```

Sets how long to keep Cache data for before invalidating it and getting new data.

#### Parameters

<i>days</i>	The number of days to store the cache data for before getting new data.
-------------	---

### 7.18.3.21 SetDebugging()

```
static void ReadingBusesAPI.ReadingBuses.SetDebugging (
    bool value ) [static]
```

Sets if you want to debug the library by making requests to a dummy server instead of the real live sever.

#### Parameters

<i>value</i>	True or False for if you want to debug or not.
--------------	--

Unless you are developing or editing library in some way you should not need to use this.

### 7.18.3.22 SetFullError()

```
static void ReadingBusesAPI.ReadingBuses.SetFullError (
    bool value ) [static]
```

Sets if you want to print out the full error logs to console, only needed for debugging library errors. Only done so in debug.

## Parameters

<i>value</i>	True or False for printing full error logs to console.
--------------	--

### 7.18.3.23 Setup()

```
async Task ReadingBusesAPI.ReadingBuses.Setup ( ) [private]
```

Creates cache data and retrieves bus services and bus stop data.

## Returns

Gets the basic bus stops and services data.

## Exceptions

<i>ReadingBusesApiExceptionBadQuery</i>	If the API key is invalid or expired.
---	---------------------------------------

### 7.18.3.24 SetWarning()

```
static void ReadingBusesAPI.ReadingBuses.SetWarning (
    bool value ) [static]
```

Sets if you want to print out warning messages to the console screen or not. Only done so in debug.

## Parameters

<i>value</i>	True or False for printing warning messages.
--------------	--

## 7.18.4 Member Data Documentation

### 7.18.4.1 \_instance

```
ReadingBuses ReadingBusesAPI.ReadingBuses._instance [static], [private]
```

The singleton instance

#### 7.18.4.2 `_locations`

```
Dictionary<string, BusStop> ReadingBusesAPI.ReadingBuses._locations [private]
```

Holds information on all the bus stops/locations visited by Reading Buses

#### 7.18.4.3 `_services`

```
List<BusService> ReadingBusesAPI.ReadingBuses._services [private]
```

Holds information on all the services operated by Reading Buses

### 7.18.5 Property Documentation

#### 7.18.5.1 `ApiKey`

```
string ReadingBusesAPI.ReadingBuses.ApiKey [static], [get], [private set], [package]
```

Holds the users API Key.

#### 7.18.5.2 `Cache`

```
bool ReadingBusesAPI.ReadingBuses.Cache = true [static], [get], [private set], [package]
```

Keeps track of if cache data is being used or not

#### 7.18.5.3 `CacheValidityLength`

```
int ReadingBusesAPI.ReadingBuses.CacheValidityLength = 7 [static], [get], [private set],  
[package]
```

Stores how many days cache data is valid for in days before being regenerated

#### 7.18.5.4 `Debugging`

```
bool ReadingBusesAPI.ReadingBuses.Debugging [static], [get], [private set], [package]
```

Keeps track of if cache data is being used or not

### 7.18.5.5 FullError

```
bool ReadingBusesAPI.ReadingBuses.FullError [static], [get], [private set], [package]
```

Keeps track of if full error logs are being outputted to console or not.

### 7.18.5.6 GpsController

```
GpsController ReadingBusesAPI.ReadingBuses.GpsController [get]
```

Stores the GPS controller, which can help get vehicle GPS data.

### 7.18.5.7 Warning

```
bool ReadingBusesAPI.ReadingBuses.Warning = true [static], [get], [private set], [package]
```

Keeps track of if warnings are being outputted to console or not.

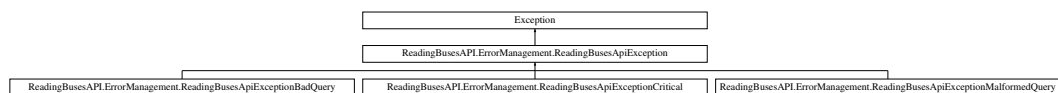
The documentation for this class was generated from the following file:

- [ReadingBuses.cs](#)

## 7.19 ReadingBusesAPI.ErrorManagement.ReadingBusesApiException Class Reference

Stores the basic/base type of Exception which can be thrown by the API.

Inheritance diagram for ReadingBusesAPI.ErrorManagement.ReadingBusesApiException:



### Package Functions

- [ReadingBusesApiException](#) ()
- [ReadingBusesApiException](#) (string content)
- [ReadingBusesApiException](#) (string message, Exception innerException)

### 7.19.1 Detailed Description

Stores the basic/base type of Exception which can be thrown by the API.

## 7.19.2 Constructor & Destructor Documentation

### 7.19.2.1 ReadingBusesApiException() [1/3]

`ReadingBusesAPI.ErrorManagement.ReadingBusesApiException.ReadingBusesApiException ( ) [package]`

### 7.19.2.2 ReadingBusesApiException() [2/3]

`ReadingBusesAPI.ErrorManagement.ReadingBusesApiException.ReadingBusesApiException ( string content ) [package]`

### 7.19.2.3 ReadingBusesApiException() [3/3]

`ReadingBusesAPI.ErrorManagement.ReadingBusesApiException.ReadingBusesApiException ( string message, Exception innerException ) [package]`

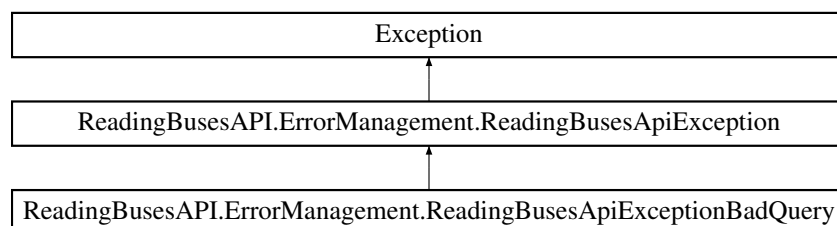
The documentation for this class was generated from the following file:

- [ErrorManagement/ReadingBusesAPIException.cs](#)

## 7.20 ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery Class Reference

An exception type which is used when the API returns back an error message. Most likely due to an invalid request such as asking for data that does not exist.

Inheritance diagram for `ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery`:





## Package Functions

- [ReadingBusesApiExceptionBadQuery](#) ([ErrorFormat](#) content)
- [ReadingBusesApiExceptionBadQuery](#) (string content)
- [ReadingBusesApiExceptionBadQuery](#) (string message, Exception innerException)

### 7.20.1 Detailed Description

An exception type which is used when the API returns back an error message. Most likely due to an invalid request such as asking for data that does not exist.

### 7.20.2 Constructor & Destructor Documentation

#### 7.20.2.1 ReadingBusesApiExceptionBadQuery() [1/3]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery.ReadingBusesApiExceptionBadQuery (
    ErrorFormat content ) [package]
```

#### 7.20.2.2 ReadingBusesApiExceptionBadQuery() [2/3]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery.ReadingBusesApiExceptionBadQuery (
    string content ) [package]
```

#### 7.20.2.3 ReadingBusesApiExceptionBadQuery() [3/3]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery.ReadingBusesApiExceptionBadQuery (
    string message,
    Exception innerException ) [package]
```

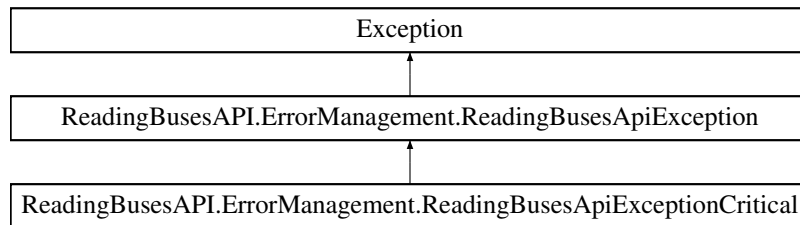
The documentation for this class was generated from the following file:

- ErrorManagement/[ReadingBusesApiExceptionBadQuery.cs](#)

## 7.21 ReadingBusesAPI.ErrorManagement.ReadingBusesApiException↔ Critical Class Reference

An exception type which is thrown when the cause of the error is unknown.

Inheritance diagram for ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical:



### Package Functions

- [ReadingBusesApiExceptionCritical](#) ()
- [ReadingBusesApiExceptionCritical](#) (string message)
- [ReadingBusesApiExceptionCritical](#) (string message, Exception innerException)

#### 7.21.1 Detailed Description

An exception type which is thrown when the cause of the error is unknown.

#### 7.21.2 Constructor & Destructor Documentation

##### 7.21.2.1 ReadingBusesApiExceptionCritical() [1/3]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical.ReadingBusesApiException↔
Critical ( ) [package]
```

##### 7.21.2.2 ReadingBusesApiExceptionCritical() [2/3]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical.ReadingBusesApiException↔
Critical (
    string message ) [package]
```

### 7.21.2.3 ReadingBusesApiExceptionCritical() [3/3]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical.ReadingBusesApiException↵
Critical (
    string message,
    Exception innerException ) [package]
```

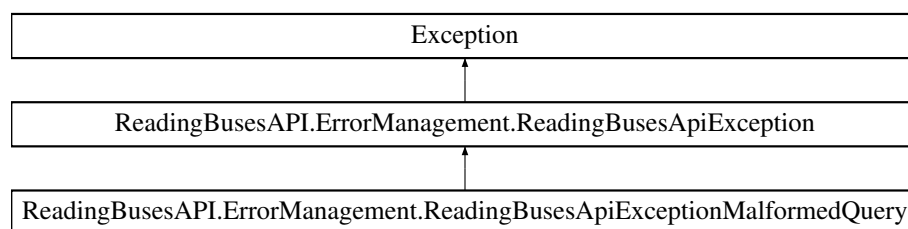
The documentation for this class was generated from the following file:

- ErrorManagement/[ReadingBusesApiExceptionCritical.cs](#)

## 7.22 ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery Class Reference

An exception type which is used when the user asks to make a invalid API call This is would be thrown during checks done before even directly calling upon the web API. For example if you have not filtered by at least one property when required too.

Inheritance diagram for ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery:



### Package Functions

- [ReadingBusesApiExceptionMalformedQuery](#) (string content)
- [ReadingBusesApiExceptionMalformedQuery](#) (string message, Exception innerException)

### 7.22.1 Detailed Description

An exception type which is used when the user asks to make a invalid API call This is would be thrown during checks done before even directly calling upon the web API. For example if you have not filtered by at least one property when required too.

### 7.22.2 Constructor & Destructor Documentation

### 7.22.2.1 ReadingBusesApiExceptionMalformedQuery() [1/2]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery.ReadingBusesApiException↵
MalformedQuery (
    string content ) [package]
```

### 7.22.2.2 ReadingBusesApiExceptionMalformedQuery() [2/2]

```
ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery.ReadingBusesApiException↵
MalformedQuery (
    string message,
    Exception innerException ) [package]
```

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

- ErrorManagement/[ReadingBusesApiExceptionMalformedQuery.cs](#)

## 7.23 ReadingBusesAPI.BusServices.Services Class Reference

This classes simply gets all the bus services operated by Reading Buses, by interfacing with the "List Of Services" API.

### Package Functions

- async Task<List<[BusService](#)>> [FindServices](#) ()  
*Finds all the services operated by Reading Buses.*

### Static Private Attributes

- const string [CacheLocation](#) = "cache\\Services.cache"  
*the location for the service cache file.*

### 7.23.1 Detailed Description

This classes simply gets all the bus services operated by Reading Buses, by interfacing with the "List Of Services" API.

### 7.23.2 Member Function Documentation

#### 7.23.2.1 FindServices()

```
async Task<List<BusService>> ReadingBusesAPI.BusServices.Services.FindServices ( ) [package]
```

Finds all the services operated by Reading Buses.

## Exceptions

<i>ReadingBusesApiException</i>	Thrown if an invalid or expired API Key is used.
---------------------------------	--

### 7.23.3 Member Data Documentation

#### 7.23.3.1 CacheLocation

```
const string ReadingBusesAPI.BusServices.Services.CacheLocation = "cache\\Services.cache"
[static], [private]
```

the location for the service cache file.

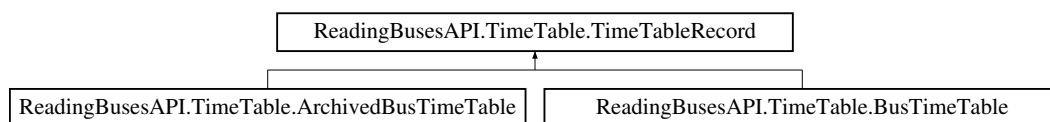
The documentation for this class was generated from the following file:

- BusServices/[Services.cs](#)

## 7.24 ReadingBusesAPI.TimeTable.TimeTableRecord Class Reference

Represents the Raw timetable object data you get from the Timetabled Journeys and Tracking History APIs.

Inheritance diagram for ReadingBusesAPI.TimeTable.TimeTableRecord:



### Classes

- class [ParseBusStopConverter](#)  
Converts a bus stop act-code into a 'BusStop' Object and back again for the JSON converter.
- class [ParseDirectionConverter](#)  
Converts a string into a Direction Enum and back again for the JSON converter.
- class [ParseTimingPointConverter](#)  
Converts a string into a boolean and back again for the JSON converter.

### Public Member Functions

- [BusService GetService \(\)](#)  
Gets the related 'BusService' object relating to the time table record.

## Package Functions

- [TimeTableRecord](#) ()

*Default constructor, to block creating objects directly.*

## Properties

- string [ServiceNumber](#) [get, set]  
*The service number of the bus.*
- [Company Operator](#) [get, set]  
*The operator of the bus services*
- [BusStop Location](#) [get, set]  
*The 'BusStop' object for the stop relating to the time table record..*
- long [Sequence](#) [get, set]  
*What number bus stop is this in the buses route, ie 1, is the first stop to visit.*
- [Direction Direction](#) [get, set]  
*Is this bus heading inbound or outbound.*
- string [JourneyCode](#) [get, set]
- bool [IsTimingPoint](#) [get, set]
- DateTime [SchArrivalTime](#) [get, set]  
*The scheduled arrival time for the bus.*
- DateTime [SchDepartureTime](#) [get, set]  
*The scheduled departure time for the bus.*

### 7.24.1 Detailed Description

Represents the Raw timetable object data you get from the Timetabled Journeys and Tracking History APIs.

### 7.24.2 Constructor & Destructor Documentation

#### 7.24.2.1 TimeTableRecord()

```
ReadingBusesAPI.TimeTable.TimeTableRecord.TimeTableRecord ( ) [package]
```

Default constructor, to block creating objects directly.

### 7.24.3 Member Function Documentation

### 7.24.3.1 GetService()

```
BusService ReadingBusesAPI.TimeTable.TimeTableRecord.GetService ( )
```

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

#### Returns

A 'BusService' object for this time table record.

## 7.24.4 Property Documentation

### 7.24.4.1 Direction

```
Direction ReadingBusesAPI.TimeTable.TimeTableRecord.Direction [get], [set]
```

Is this bus heading inbound or outbound.

### 7.24.4.2 IsTimingPoint

```
bool ReadingBusesAPI.TimeTable.TimeTableRecord.IsTimingPoint [get], [set]
```

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.

### 7.24.4.3 JourneyCode

```
string ReadingBusesAPI.TimeTable.TimeTableRecord.JourneyCode [get], [set]
```

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.

### 7.24.4.4 Location

```
BusStop ReadingBusesAPI.TimeTable.TimeTableRecord.Location [get], [set]
```

The 'BusStop' object for the stop relating to the time table record..

#### 7.24.4.5 Operator

`Company ReadingBusesAPI.TimeTable.TimeTableRecord.Operator [get], [set]`

The operator of the bus services

#### 7.24.4.6 SchArrivalTime

`DateTime ReadingBusesAPI.TimeTable.TimeTableRecord.SchArrivalTime [get], [set]`

The scheduled arrival time for the bus.

#### 7.24.4.7 SchDepartureTime

`DateTime ReadingBusesAPI.TimeTable.TimeTableRecord.SchDepartureTime [get], [set]`

The scheduled departure time for the bus.

#### 7.24.4.8 Sequence

`long ReadingBusesAPI.TimeTable.TimeTableRecord.Sequence [get], [set]`

What number bus stop is this in the buses route, ie 1, is the first stop to visit.

#### 7.24.4.9 ServiceNumber

`string ReadingBusesAPI.TimeTable.TimeTableRecord.ServiceNumber [get], [set], [package]`

The service number of the bus.

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

- [TimeTable/TimeTableRecord.cs](#)

## 7.25 ReadingBusesAPI.Common.UrlConstructor Class Reference

Returns back the URL needed to make a get command to the Reading Buses Open Data API. You can use this for testing purposes to check the API is returning what you were expecting.



## Static Public Member Functions

- static string [ListOfBusStops](#) ()  
*Returns back the URL needed for a get request to the 'List of Bus Stops' API.*
- static string [LiveVehiclePositions](#) ()  
*Returns back the URL needed for a get request to the 'Live Vehicle Positions' API.*
- static string [LiveJourneyDetails](#) ()  
*Returns back the URL needed for a get request to the 'Live Journey Details' API.*
- static string [StopPredictions](#) (string actoCode)  
*Returns back the URL needed for a get request to the 'Stop Predictions' API.*
- static string [ListOfServices](#) ()  
*Returns back the URL needed for a get request to the 'List of Services' API.*
- static string [LinePatterns](#) ([BusService](#) service)  
*Returns back the URL needed for a get request to the 'Line Patterns' API.*
- static string [TimetabledJourneys](#) ([BusService](#) service, [BusStop](#) location, DateTime date)  
*Returns back the URL needed for a get request to the 'Time Tabled Journeys' API.*
- static string [TrackingHistory](#) ([BusService](#) service, [BusStop](#) location, DateTime date, string vehicle)  
*Returns back the URL needed for a get request to the 'Tracking History' API.*
- static string [VehiclePositionHistory](#) (DateTime dateStartTime, TimeSpan? timeSpan, string vehicle)  
*Returns back the URL needed for a get request to the 'Vehicle Position History' API.*

## Static Private Member Functions

- static DateTime [AddTimeSpan](#) (DateTime start, TimeSpan? timeSpan)  
*Adds the time span onto the start date time. If the time span expands into the next day stop it and limit it to today only. If no time span was given then assume they want a full day of data.*

## Static Private Attributes

- const string [ReadingBusesApi](#) = "https://rtl2.ods-live.co.uk/api/"  
*URL for the Reading buses Open Data API server.*
- const string [DummyApi](#) = "https://jonathanfoot.com/Projects/RBAPI/test-data/"  
*URL for the Reading buses Open Data API server.*

### 7.25.1 Detailed Description

Returns back the URL needed to make a get command to the Reading Buses Open Data API. You can use this for testing purposes to check the API is returning what you were expecting.

### 7.25.2 Member Function Documentation

#### 7.25.2.1 AddTimeSpan()

```
static DateTime ReadingBusesAPI.Common.UrlConstructor.AddTimeSpan (
    DateTime start,
    TimeSpan? timeSpan ) [static], [private]
```

Adds the time span onto the start date time. If the time span expands into the next day stop it and limit it to today only. If no time span was given then assume they want a full day of data.

**Parameters**

<i>start</i>	The start date time, for what day and what time they want to get data from.
<i>timeSpan</i>	The length of time you want data for. Must be a positive value. When added to 'start' it should not take you into the next day.

**Returns**

The new DateTime object, which has the start date time object incremented by the time span safely.

**7.25.2.2 LinePatterns()**

```
static string ReadingBusesAPI.Common.UrlConstructor.LinePatterns (
    BusService service ) [static]
```

Returns back the URL needed for a get request to the 'Line Patterns' API.

**Parameters**

<i>service</i>	The bus service to query for.
----------------	-------------------------------

**Returns**

Returns back the URL needed for a get request to the 'Line Patterns' API.

**7.25.2.3 ListOfBusStops()**

```
static string ReadingBusesAPI.Common.UrlConstructor.ListOfBusStops ( ) [static]
```

Returns back the URL needed for a get request to the 'List of Bus Stops' API.

**Returns**

Returns back the URL needed for a get request to the 'List of Bus Stops' API.

**7.25.2.4 ListOfServices()**

```
static string ReadingBusesAPI.Common.UrlConstructor.ListOfServices ( ) [static]
```

Returns back the URL needed for a get request to the 'List of Services' API.

**Returns**

Returns back the URL needed for a get request to the 'List of Services' API.

#### 7.25.2.5 LiveJourneyDetails()

```
static string ReadingBusesAPI.Common.UrlConstructor.LiveJourneyDetails ( ) [static]
```

Returns back the URL needed for a get request to the 'Live Journey Details' API.

##### Returns

Returns back the URL needed for a get request to the 'Live Journey Details' API.

#### 7.25.2.6 LiveVehiclePositions()

```
static string ReadingBusesAPI.Common.UrlConstructor.LiveVehiclePositions ( ) [static]
```

Returns back the URL needed for a get request to the 'Live Vehicle Positions' API.

##### Returns

Returns back the URL needed for a get request to the 'Live Vehicle Positions' API.

#### 7.25.2.7 StopPredictions()

```
static string ReadingBusesAPI.Common.UrlConstructor.StopPredictions (
    string actoCode ) [static]
```

Returns back the URL needed for a get request to the 'Stop Predictions' API.

##### Parameters

<i>actoCode</i>	The bus stop ID code.
-----------------	-----------------------

##### Returns

Returns back the URL needed for a get request to the 'Stop Predictions' API.

#### 7.25.2.8 TimetabledJourneys()

```
static string ReadingBusesAPI.Common.UrlConstructor.TimetabledJourneys (
    BusService service,
    BusStop location,
    DateTime date ) [static]
```

Returns back the URL needed for a get request to the 'Time Tabled Journeys' API.

**Parameters**

<i>service</i>	The bus service to query for.
<i>location</i>	The bus stop to query for.
<i>date</i>	The date you want to query at.

**Returns**

Returns back the URL needed for a get request to the 'Time Tabled Journeys' API.

**7.25.2.9 TrackingHistory()**

```
static string ReadingBusesAPI.Common.UrlConstructor.TrackingHistory (
    BusService service,
    BusStop location,
    DateTime date,
    string vehicle ) [static]
```

Returns back the URL needed for a get request to the 'Tracking History' API.

**Parameters**

<i>service</i>	The bus service to query for.
<i>location</i>	The bus stop to query for.
<i>date</i>	The date you want to query at.
<i>vehicle</i>	A vehicle ID to query for.

**Returns**

Returns back the URL needed for a get request to the 'Tracking History' API.

**7.25.2.10 VehiclePositionHistory()**

```
static string ReadingBusesAPI.Common.UrlConstructor.VehiclePositionHistory (
    DateTime dateStartTime,
    TimeSpan? timeSpan,
    string vehicle ) [static]
```

Returns back the URL needed for a get request to the 'Vehicle Position History' API.

**Parameters**

<i>dateStartTime</i>	The date and time you want to query for.
<i>timeSpan</i>	The length of the time period to query for.
<i>vehicle</i>	A vehicle ID to query for.

### Returns

Returns back the URL needed for a get request to the 'Vehicle Position History' API.

## 7.25.3 Member Data Documentation

### 7.25.3.1 DummyApi

```
const string ReadingBusesAPI.Common.UrlConstructor.DummyApi = "https://jonathanfoot.com/Projects/R↳  
BAPI/test-data/" [static], [private]
```

URL for the Reading buses Open Data API server.

### 7.25.3.2 ReadingBusesApi

```
const string ReadingBusesAPI.Common.UrlConstructor.ReadingBusesApi = "https://rtl2.ods-live.↳  
co.uk/api/" [static], [private]
```

URL for the Reading buses Open Data API server.

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

- [Common/URLConstructor.cs](#)



## Chapter 8

# File Documentation

### 8.1 BusServices/BusService.cs File Reference

#### Classes

- class [ReadingBusesAPI.BusServices.BusService](#)  
*Stores information about an individual bus services. Related to the "List Of Services" API.*

#### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.BusServices](#)

### 8.2 BusServices/Services.cs File Reference

#### Classes

- class [ReadingBusesAPI.BusServices.Services](#)  
*This classes simply gets all the bus services operated by Reading Buses, by interfacing with the "List Of Services" API.*

#### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.BusServices](#)

### 8.3 BusStops/BusStop.cs File Reference

#### Classes

- class [ReadingBusesAPI.BusStops.BusStop](#)  
*Stores information about a single bus stop. Related to the "List Of Bus Stops" API.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.BusStops](#)

## 8.4 BusStops/Locations.cs File Reference

### Classes

- class [ReadingBusesAPI.BusStops.Locations](#)

*This classes simply gets all the buses stops visited by Reading Buses, by interfacing with the "List Of Bus Stops" API.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.BusStops](#)

## 8.5 Common/Direction.cs File Reference

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.Common](#)

### Enumerations

- enum [ReadingBusesAPI.Common.Direction](#) { [ReadingBusesAPI.Common.Direction.Inbound](#), [ReadingBusesAPI.Common.Direction.Outbound](#) }

*The direction in which a bus is traveling.*

## 8.6 Common/JSONConverters.cs File Reference

### Classes

- class [ReadingBusesAPI.Common.ParseOperatorConverter](#)

*Converts a string short code for an Operator into an Operator Enum and back again for the JSON converter.*

- class [ReadingBusesAPI.Common.ParseStringConverter](#)

*Converts a string into a long and back again for the JSON converter.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.Common](#)



## 8.7 Common/Operators.cs File Reference

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.Common](#)

### Enumerations

- enum [ReadingBusesAPI.Common.Company](#) { [ReadingBusesAPI.Common.Company.ReadingBuses](#), [ReadingBusesAPI.Common.Company.Kennections](#), [ReadingBusesAPI.Common.Company.NewburyAndDistrict](#), [ReadingBusesAPI.Common.Company.Other](#) }

*An Enum of the Operators Reading Buses owns or manages in their API.*

## 8.8 Common/Point.cs File Reference

### Classes

- struct [ReadingBusesAPI.Common.Point](#)

*Stores an X and Y Position simply.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.Common](#)

## 8.9 Common/URLConstructor.cs File Reference

### Classes

- class [ReadingBusesAPI.Common.UrlConstructor](#)

*Returns back the URL needed to make a get command to the Reading Buses Open Data API. You can use this for testing purposes to check the API is returning what you were expecting.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.Common](#)

## 8.10 ErrorManagement/ErrorFormat.cs File Reference

### Classes

- class [ReadingBusesAPI.ErrorManagement.ErrorFormat](#)

*Represents an error message object returned by all JSON feeds of the API.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.ErrorManagement](#)

## 8.11 ErrorManagement/ErrorMessage.cs File Reference

### Classes

- class [ReadingBusesAPI.ErrorManagement.ErrorMessage](#)  
*Responsible for extracting and producing an error message from an API result. To the end user.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.ErrorManagement](#)

## 8.12 ErrorManagement/ReadingBusesAPIException.cs File Reference

### Classes

- class [ReadingBusesAPI.ErrorManagement.ReadingBusesAPIException](#)  
*Stores the basic/base type of Exception which can be thrown by the API.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.ErrorManagement](#)

## 8.13 ErrorManagement/ReadingBusesAPIExceptionBadRequest.cs File Reference

### Classes

- class [ReadingBusesAPI.ErrorManagement.ReadingBusesAPIExceptionBadRequest](#)  
*An exception type which is used when the API returns back an error message. Most likely due to an invalid request such as asking for data that does not exist.*

## Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.ErrorManagement](#)

## 8.14 ErrorManagement/ReadingBusesApiExceptionCritical.cs File Reference

### Classes

- class [ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical](#)

*An exception type which is thrown when the cause of the error is unknown.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.ErrorManagement](#)

## 8.15 ErrorManagement/ReadingBusesApiExceptionMalformedQuery.cs File Reference

### Classes

- class [ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery](#)

*An exception type which is used when the user asks to make a invalid API call This is would be thrown during checks done before even directly calling upon the web API. For example if you have not filtered by at least one property when required too.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.ErrorManagement](#)

## 8.16 JourneyDetails/LiveRecord.cs File Reference

### Classes

- class [ReadingBusesAPI.JourneyDetails.LiveRecord](#)

*Used to store information about a buses arrival at a bus stop. Mainly related to the "Stop Predictions" API.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.JourneyDetails](#)

## 8.17 ReadingBuses.cs File Reference

### Classes

- class [ReadingBusesAPI.ReadingBuses](#)

*This is the main class for the library, here you can initialise a singleton instance and then query and use the Reading Buses API.*

### Namespaces

- namespace [ReadingBusesAPI](#)

## 8.18 README.md File Reference

## 8.19 TimeTable/ArchivedBusTimeTable.cs File Reference

### Classes

- class [ReadingBusesAPI.TimeTable.ArchivedBusTimeTable](#)

*Represents and retrieves information about a actual single time table record, which means information on one bus at one location. Related to the "Tracking History" API.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.TimeTable](#)

## 8.20 TimeTable/BusTimeTable.cs File Reference

### Classes

- class [ReadingBusesAPI.TimeTable.BusTimeTable](#)

*Represents and retrieves information about a scheduled/predicted single time table record, which means information on one bus at one location. Related to the "Timetabled Journeys" API.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.TimeTable](#)

## 8.21 TimeTable/TimeTableRecord.cs File Reference

### Classes

- class [ReadingBusesAPI.TimeTable.TimeTableRecord](#)  
*Represents the Raw timetable object data you get from the Timetabled Journeys and Tracking History APIs.*
- class [ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter](#)  
*Converts a bus stop actio-code into a 'BusStop' Object and back again for the JSON converter.*
- class [ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter](#)  
*Converts a string into a Direction Enum and back again for the JSON converter.*
- class [ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter](#)  
*Converts a string into a boolean and back again for the JSON converter.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.TimeTable](#)

## 8.22 VehiclePositions/ArchivedPositions.cs File Reference

### Classes

- class [ReadingBusesAPI.VehiclePositions.ArchivedPositions](#)  
*Stores information about previous/ archived GPS data on vehicles.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.VehiclePositions](#)

## 8.23 VehiclePositions/GPSController.cs File Reference

### Classes

- class [ReadingBusesAPI.VehiclePositions.GpsController](#)  
*Helps get live and historical GPS data on vehicles by accessing the "Live Vehicle Positions" API.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.VehiclePositions](#)

## 8.24 VehiclePositions/LivePosition.cs File Reference

### Classes

- class [ReadingBusesAPI.VehiclePositions.LivePosition](#)  
*Used to store live information about a buses GPS position. Related to the "Live Vehicle Positions" API.*

### Namespaces

- namespace [ReadingBusesAPI](#)
- namespace [ReadingBusesAPI.VehiclePositions](#)



# Index

- [\\_instance](#)
    - [ReadingBusesAPI.ReadingBuses, 73](#)
  - [\\_lastRetrieval](#)
    - [ReadingBusesAPI.VehiclePositions.GpsController, 44](#)
  - [\\_livePositionCache](#)
    - [ReadingBusesAPI.VehiclePositions.GpsController, 44](#)
  - [\\_locations](#)
    - [ReadingBusesAPI.ReadingBuses, 73](#)
  - [\\_services](#)
    - [ReadingBusesAPI.ReadingBuses, 74](#)
  - [\\_stops](#)
    - [ReadingBusesAPI.BusServices.BusService, 28](#)
  - [\\_stopsObjects](#)
    - [ReadingBusesAPI.BusServices.BusService, 28](#)
- [ActArrivalTime](#)
  - [ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 18](#)
- [ActDepartureTime](#)
  - [ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 18](#)
- [ActoCode](#)
  - [ReadingBusesAPI.BusStops.BusStop, 35](#)
- [AddTimeSpan](#)
  - [ReadingBusesAPI.Common.UrlConstructor, 85](#)
- [ApiKey](#)
  - [ReadingBusesAPI.ReadingBuses, 74](#)
- [ArchivedBusTimeTable](#)
  - [ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 16](#)
- [ArchivedPositions](#)
  - [ReadingBusesAPI.VehiclePositions.ArchivedPositions, 19](#)
- [ArrivalLateness](#)
  - [ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 16](#)
- [ArrivalMin](#)
  - [ReadingBusesAPI.JourneyDetails.LiveRecord, 47](#)
- [Bearing](#)
  - [ReadingBusesAPI.BusStops.BusStop, 35](#)
  - [ReadingBusesAPI.VehiclePositions.LivePosition, 46](#)
- [BrandName](#)
  - [ReadingBusesAPI.BusServices.BusService, 29](#)
- [BusService](#)
  - [ReadingBusesAPI.BusServices.BusService, 22, 23](#)
- [BusServices/BusService.cs, 91](#)
- [BusServices/Services.cs, 91](#)
- [BusStop](#)
  - [ReadingBusesAPI.BusStops.BusStop, 30, 31](#)
- [BusStops/BusStop.cs, 91](#)
- [BusStops/Locations.cs, 92](#)
- [BusTimeTable](#)
  - [ReadingBusesAPI.TimeTable.BusTimeTable, 37](#)
- [Cache](#)
  - [ReadingBusesAPI.ReadingBuses, 74](#)
- [CacheLocation](#)
  - [ReadingBusesAPI.BusServices.Services, 81](#)
  - [ReadingBusesAPI.BusStops.Locations, 51](#)
- [CacheValidityLength](#)
  - [ReadingBusesAPI.ReadingBuses, 74](#)
- [CanConvert](#)
  - [ReadingBusesAPI.Common.ParseOperatorConverter, 55](#)
  - [ReadingBusesAPI.Common.ParseStringConverter, 56](#)
  - [ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter, 52](#)
  - [ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter, 53](#)
  - [ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter, 58](#)
- [Code](#)
  - [ReadingBusesAPI.ErrorManagement.ErrorFormat, 39](#)
  - [Common/Direction.cs, 92](#)
  - [Common/JSONConverters.cs, 92](#)
  - [Common/Operators.cs, 93](#)
  - [Common/Point.cs, 93](#)
  - [Common/URLConstructor.cs, 93](#)
- [CommonName](#)
  - [ReadingBusesAPI.BusStops.BusStop, 35](#)
- [Company](#)
  - [ReadingBusesAPI.Common, 12](#)
- [Debugging](#)
  - [ReadingBusesAPI.ReadingBuses, 74](#)
- [DeleteInstance](#)
  - [ReadingBusesAPI.ReadingBuses, 65](#)
- [DepartureLateness](#)
  - [ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 16](#)
- [Destination](#)
  - [ReadingBusesAPI.JourneyDetails.LiveRecord, 49](#)
- [Direction](#)

- ReadingBusesAPI.Common, 12
- ReadingBusesAPI.TimeTable.TimeTableRecord, 83
- DisplayTime
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 48
- DummyApi
  - ReadingBusesAPI.Common.UrlConstructor, 89
- Equals
  - ReadingBusesAPI.Common.Point, 60
- ErrorManagement/ErrorFormat.cs, 93
- ErrorManagement/ErrorMessageManager.cs, 94
- ErrorManagement/ReadingBusesAPIException.cs, 94
- ErrorManagement/ReadingBusesAPIExceptionBadQuery.cs, 94
- ErrorManagement/ReadingBusesAPIExceptionCritical.cs, 95
- ErrorManagement/ReadingBusesAPIExceptionMalformedQuery.cs, 95
- ExptArrival
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 49
- FindLocations
  - ReadingBusesAPI.BusStops.Locations, 50
- FindServices
  - ReadingBusesAPI.BusServices.Services, 80
- FullError
  - ReadingBusesAPI.ReadingBuses, 74
- GetArchivedTimeTable
  - ReadingBusesAPI.BusServices.BusService, 23, 24
  - ReadingBusesAPI.BusStops.BusStop, 31
- GetArchivedVehiclePositions
  - ReadingBusesAPI.VehiclePositions.GpsController, 41, 42
- GetGroupedArchivedTimeTable
  - ReadingBusesAPI.BusServices.BusService, 24, 25
- GetGroupedTimeTable
  - ReadingBusesAPI.BusServices.BusService, 25, 26
  - ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 16
  - ReadingBusesAPI.TimeTable.BusTimeTable, 37
- GetHashCode
  - ReadingBusesAPI.Common.Point, 60
- GetInstance
  - ReadingBusesAPI.ReadingBuses, 65
- GetLiveData
  - ReadingBusesAPI.BusStops.BusStop, 33
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 48
- GetLivePositions
  - ReadingBusesAPI.BusServices.BusService, 26
- GetLiveVehiclePosition
  - ReadingBusesAPI.VehiclePositions.GpsController, 43
- GetLiveVehiclePositions
  - ReadingBusesAPI.VehiclePositions.GpsController, 43
- GetLocation
  - ReadingBusesAPI.ReadingBuses, 65
- GetLocations
  - ReadingBusesAPI.BusServices.BusService, 26
  - ReadingBusesAPI.ReadingBuses, 66
- GetLocationsActo
  - ReadingBusesAPI.BusServices.BusService, 27
- GetOperatorE
  - ReadingBusesAPI.ReadingBuses, 66
- GetPoint
  - ReadingBusesAPI.BusStops.BusStop, 33
  - ReadingBusesAPI.VehiclePositions.ArchivedPositions, 19
- GetService
  - ReadingBusesAPI.ReadingBuses, 67
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 82
  - ReadingBusesAPI.VehiclePositions.LivePosition, 46
- GetServices
  - ReadingBusesAPI.BusStops.BusStop, 33
  - ReadingBusesAPI.ReadingBuses, 68
- GetStops
  - ReadingBusesAPI.BusServices.BusService, 27
- GetTimeTable
  - ReadingBusesAPI.BusServices.BusService, 27, 28
  - ReadingBusesAPI.BusStops.BusStop, 34
  - ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, 17
  - ReadingBusesAPI.TimeTable.BusTimeTable, 38
- GetVehicleTrackingHistory
  - ReadingBusesAPI.ReadingBuses, 69
- GpsController
  - ReadingBusesAPI.ReadingBuses, 75
  - ReadingBusesAPI.VehiclePositions.GpsController, 41
- GroupName
  - ReadingBusesAPI.BusStops.BusStop, 35
- Inbound
  - ReadingBusesAPI.Common, 13
- Initialise
  - ReadingBusesAPI.ReadingBuses, 69
- InvalidateCache
  - ReadingBusesAPI.ReadingBuses, 70
- IsCacheValid
  - ReadingBusesAPI.VehiclePositions.GpsController, 43
- IsLocation
  - ReadingBusesAPI.ReadingBuses, 70
- IsService
  - ReadingBusesAPI.ReadingBuses, 70
- IsTimingPoint
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 83
- IsVehicle



- ReadingBusesAPI.VehiclePositions.GpsController, 44
- JourneyCode
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 83
- JourneyDetails/LiveRecord.cs, 95
- Kennections
  - ReadingBusesAPI.Common, 12
- Latitude
  - ReadingBusesAPI.BusStops.BusStop, 35
  - ReadingBusesAPI.VehiclePositions.ArchivedPositions, 20
- LinePatterns
  - ReadingBusesAPI.Common.UrlConstructor, 86
- ListOfBusStops
  - ReadingBusesAPI.Common.UrlConstructor, 86
- ListOfServices
  - ReadingBusesAPI.Common.UrlConstructor, 86
- LiveJourneyDetails
  - ReadingBusesAPI.Common.UrlConstructor, 86
- LivePosition
  - ReadingBusesAPI.VehiclePositions.LivePosition, 45
- LiveRecord
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 47
- LiveVehiclePositions
  - ReadingBusesAPI.Common.UrlConstructor, 87
- Location
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 83
- Longitude
  - ReadingBusesAPI.BusStops.BusStop, 35
  - ReadingBusesAPI.VehiclePositions.ArchivedPositions, 20
- Message
  - ReadingBusesAPI.ErrorManagement.ErrorFormat, 39
- NewburyAndDistrict
  - ReadingBusesAPI.Common, 12
- Observed
  - ReadingBusesAPI.VehiclePositions.ArchivedPositions, 20
- Operator
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 83
- operator!=
  - ReadingBusesAPI.Common.Point, 61
- operator==
  - ReadingBusesAPI.Common.Point, 61
- OperatorCode
  - ReadingBusesAPI.BusServices.BusService, 29
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 49
  - ReadingBusesAPI.VehiclePositions.ArchivedPositions, 20
- Other
  - ReadingBusesAPI.Common, 12
- Outbound
  - ReadingBusesAPI.Common, 13
- Point
  - ReadingBusesAPI.Common.Point, 59
- PrintFullErrorLogs
  - ReadingBusesAPI.ReadingBuses, 71
- PrintWarning
  - ReadingBusesAPI.ReadingBuses, 71
- ReadingBuses
  - ReadingBusesAPI.Common, 12
  - ReadingBusesAPI.ReadingBuses, 65
- ReadingBuses.cs, 96
- ReadingBusesAPI, 11
- ReadingBusesApi
  - ReadingBusesAPI.Common.UrlConstructor, 89
- ReadingBusesAPI.BusServices, 11
- ReadingBusesAPI.BusServices.BusService, 21
  - \_stops, 28
  - \_stopsObjects, 28
  - BrandName, 29
  - BusService, 22, 23
  - GetArchivedTimeTable, 23, 24
  - GetGroupedArchivedTimeTable, 24, 25
  - GetGroupedTimeTable, 25, 26
  - GetLivePositions, 26
  - GetLocations, 26
  - GetLocationsActo, 27
  - GetStops, 27
  - GetTimeTable, 27, 28
  - OperatorCode, 29
  - ServicId, 29
- ReadingBusesAPI.BusServices.Services, 80
  - CacheLocation, 81
  - FindServices, 80
- ReadingBusesAPI.BusStops, 11
- ReadingBusesAPI.BusStops.BusStop, 29
  - ActoCode, 35
  - Bearing, 35
  - BusStop, 30, 31
  - CommonName, 35
  - GetArchivedTimeTable, 31
  - GetLiveData, 33
  - GetPoint, 33
  - GetServices, 33
  - GetTimeTable, 34
  - GroupName, 35
  - Latitude, 35
  - Longitude, 35
  - ServicesString, 36
- ReadingBusesAPI.BusStops.Locations, 50
  - CacheLocation, 51
  - FindLocations, 50
- ReadingBusesAPI.Common, 12
  - Company, 12
  - Direction, 12

- Inbound, [13](#)
- Kennections, [12](#)
- NewburyAndDistrict, [12](#)
- Other, [12](#)
- Outbound, [13](#)
- ReadingBuses, [12](#)
- ReadingBusesAPI.Common.ParseOperatorConverter, [54](#)
  - CanConvert, [55](#)
  - ReadJson, [55](#)
  - Singleton, [55](#)
  - WriteJson, [55](#)
- ReadingBusesAPI.Common.ParseStringConverter, [56](#)
  - CanConvert, [56](#)
  - ReadJson, [56](#)
  - Singleton, [57](#)
  - WriteJson, [56](#)
- ReadingBusesAPI.Common.Point, [59](#)
  - Equals, [60](#)
  - GetHashCode, [60](#)
  - operator!=, [61](#)
  - operator==, [61](#)
  - Point, [59](#)
  - ToString, [61](#)
  - X, [62](#)
  - Y, [62](#)
- ReadingBusesAPI.Common.UrlConstructor, [84](#)
  - AddTimeSpan, [85](#)
  - DummyApi, [89](#)
  - LinePatterns, [86](#)
  - ListOfBusStops, [86](#)
  - ListOfServices, [86](#)
  - LiveJourneyDetails, [86](#)
  - LiveVehiclePositions, [87](#)
  - ReadingBusesApi, [89](#)
  - StopPredictions, [87](#)
  - TimetabledJourneys, [87](#)
  - TrackingHistory, [88](#)
  - VehiclePositionHistory, [88](#)
- ReadingBusesAPI.ErrorManagement, [13](#)
- ReadingBusesAPI.ErrorManagement.ErrorFormat, [38](#)
  - Code, [39](#)
  - Message, [39](#)
  - Status, [39](#)
- ReadingBusesAPI.ErrorManagement.ErrorManager, [39](#)
  - TryErrorMessageRetrieval, [40](#)
- ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionWarning, [75](#)
  - ReadingBusesApiException, [76](#)
- ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery, [76](#)
  - ReadingBusesApiExceptionBadQuery, [77](#)
- ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical, [78](#)
  - ReadingBusesApiExceptionCritical, [78](#)
- ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery, [79](#)
  - ReadingBusesApiExceptionMalformedQuery, [79](#)
- ReadingBusesAPI.JourneyDetails, [13](#)
- ReadingBusesAPI.JourneyDetails.LiveRecord, [46](#)
  - ArrivalMin, [47](#)
  - Destination, [49](#)
  - DisplayTime, [48](#)
  - ExptArrival, [49](#)
  - GetLiveData, [48](#)
  - LiveRecord, [47](#)
  - OperatorCode, [49](#)
  - SchArrival, [49](#)
  - Service, [48](#)
  - ServiceNumber, [49](#)
  - VehicleRef, [50](#)
  - ViaMessage, [50](#)
- ReadingBusesAPI.ReadingBuses, [62](#)
  - \_instance, [73](#)
  - \_locations, [73](#)
  - \_services, [74](#)
  - ApiKey, [74](#)
  - Cache, [74](#)
  - CacheValidityLength, [74](#)
  - Debugging, [74](#)
  - DeleteInstance, [65](#)
  - FullError, [74](#)
  - GetInstance, [65](#)
  - GetLocation, [65](#)
  - GetLocations, [66](#)
  - GetOperatorE, [66](#)
  - GetService, [67](#)
  - GetServices, [68](#)
  - GetVehicleTrackingHistory, [69](#)
  - GpsController, [75](#)
  - Initialise, [69](#)
  - InvalidateCache, [70](#)
  - IsLocation, [70](#)
  - IsService, [70](#)
  - PrintFullErrorLogs, [71](#)
  - PrintWarning, [71](#)
  - ReadingBuses, [65](#)
  - SetCache, [71](#)
  - SetCacheValidityLength, [72](#)
  - SetDebugging, [72](#)
  - SetFullError, [72](#)
  - SetUp, [73](#)
  - SetWarning, [73](#)
  - Warning, [75](#)
- ReadingBusesAPI.TimeTable, [13](#)
- ReadingBusesAPI.TimeTable.ArchivedBusTimeTable, [16](#)
  - ActArrivalTime, [18](#)
  - ActDepartureTime, [18](#)
  - ArchivedBusTimeTable, [16](#)
  - ArrivalLateness, [16](#)
  - DepartureLateness, [16](#)
  - GetArrangedQuery, [16](#)
  - GetTimeTable, [17](#)

- ReadingBusesAPI.TimeTable.BusTimeTable, 36
  - BusTimeTable, 37
  - GetGroupedTimeTable, 37
  - GetTimeTable, 38
- ReadingBusesAPI.TimeTable.TimeTableRecord, 81
  - Direction, 83
  - GetService, 82
  - IsTimingPoint, 83
  - JourneyCode, 83
  - Location, 83
  - Operator, 83
  - SchArrivalTime, 84
  - SchDepartureTime, 84
  - Sequence, 84
  - ServiceNumber, 84
  - TimeTableRecord, 82
- ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter, 51
  - CanConvert, 52
  - ReadJson, 52
  - Singleton, 52
  - WriteJson, 52
- ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter, 53
  - CanConvert, 53
  - ReadJson, 53
  - Singleton, 54
  - WriteJson, 53
- ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter, 57
  - CanConvert, 58
  - ReadJson, 58
  - Singleton, 58
  - WriteJson, 58
- ReadingBusesAPI.VehiclePositions, 14
- ReadingBusesAPI.VehiclePositions.ArchivedPositions, 18
  - ArchivedPositions, 19
  - GetPoint, 19
  - Latitude, 20
  - Longitude, 20
  - Observed, 20
  - OperatorCode, 20
  - Vehicle, 20
- ReadingBusesAPI.VehiclePositions.GpsController, 40
  - \_lastRetrieval, 44
  - \_livePositionCache, 44
  - GetArchivedVehiclePositions, 41, 42
  - GetLiveVehiclePosition, 43
  - GetLiveVehiclePositions, 43
  - GpsController, 41
  - IsCacheValid, 43
  - IsVehicle, 44
- ReadingBusesAPI.VehiclePositions.LivePosition, 45
  - Bearing, 46
  - GetService, 46
  - LivePosition, 45
  - ServiceId, 46
- ReadingBusesApiException
  - ReadingBusesAPI.ErrorManagement.ReadingBusesApiException, 76
- ReadingBusesApiExceptionBadQuery
  - ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionBadQuery, 77
- ReadingBusesApiExceptionCritical
  - ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionCritical, 78
- ReadingBusesApiExceptionMalformedQuery
  - ReadingBusesAPI.ErrorManagement.ReadingBusesApiExceptionMalformedQuery, 79, 80
- ReadJson
  - ReadingBusesAPI.Common.ParseOperatorConverter, 55
  - ReadingBusesAPI.Common.ParseStringConverter, 56
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter, 52
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter, 53
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter, 58
- README.md, 96
- SchArrival
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 49
- SchArrivalTime
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 84
- SchDepartureTime
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 84
- Sequence
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 84
- Service
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 48
- ServiceId
  - ReadingBusesAPI.BusServices.BusService, 29
  - ReadingBusesAPI.VehiclePositions.LivePosition, 46
- ServiceNumber
  - ReadingBusesAPI.JourneyDetails.LiveRecord, 49
  - ReadingBusesAPI.TimeTable.TimeTableRecord, 84
- ServicesString
  - ReadingBusesAPI.BusStops.BusStop, 36
- SetCache
  - ReadingBusesAPI.ReadingBuses, 71
- SetCacheValidityLength
  - ReadingBusesAPI.ReadingBuses, 72
- SetDebugging
  - ReadingBusesAPI.ReadingBuses, 72
- SetFullError
  - ReadingBusesAPI.ReadingBuses, 72
- SetUp
  - ReadingBusesAPI.ReadingBuses, 73
- SetWarning

- ReadingBusesAPI.ReadingBuses, [73](#)
- Singleton
  - ReadingBusesAPI.Common.ParseOperatorConverter, [55](#)
  - ReadingBusesAPI.Common.ParseStringConverter, [57](#)
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter, [52](#)
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter, [54](#)
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseTimingPointConverter, [58](#)
- Status
  - ReadingBusesAPI.ErrorManagement.ErrorFormat, [39](#)
- StopPredictions
  - ReadingBusesAPI.Common.UrlConstructor, [87](#)
- TimeTable/ArchivedBusTimeTable.cs, [96](#)
- TimeTable/BusTimeTable.cs, [96](#)
- TimeTable/TimeTableRecord.cs, [97](#)
- TimetabledJourneys
  - ReadingBusesAPI.Common.UrlConstructor, [87](#)
- TimeTableRecord
  - ReadingBusesAPI.TimeTable.TimeTableRecord, [82](#)
- ToString
  - ReadingBusesAPI.Common.Point, [61](#)
- TrackingHistory
  - ReadingBusesAPI.Common.UrlConstructor, [88](#)
- TryErrorMessageRetrieval
  - ReadingBusesAPI.ErrorManagement.ErrorManager, [40](#)
- Vehicle
  - ReadingBusesAPI.VehiclePositions.ArchivedPositions, [20](#)
- VehiclePositionHistory
  - ReadingBusesAPI.Common.UrlConstructor, [88](#)
- VehiclePositions/ArchivedPositions.cs, [97](#)
- VehiclePositions/GPSController.cs, [97](#)
- VehiclePositions/LivePosition.cs, [97](#)
- VehicleRef
  - ReadingBusesAPI.JourneyDetails.LiveRecord, [50](#)
- ViaMessage
  - ReadingBusesAPI.JourneyDetails.LiveRecord, [50](#)
- Warning
  - ReadingBusesAPI.ReadingBuses, [75](#)
- WriteJson
  - ReadingBusesAPI.Common.ParseOperatorConverter, [55](#)
  - ReadingBusesAPI.Common.ParseStringConverter, [56](#)
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseBusStopConverter, [52](#)
  - ReadingBusesAPI.TimeTable.TimeTableRecord.ParseDirectionConverter, [53](#)