Version number: **2.1**

Date: **25 October 2011**

Traveline API – Guidance for Developers

Prepared by:

Traveline & MxData

Version control

Ver.	Date	Author	Description
2.0	06/10/11	ML	New Supplier + Add Appendix 2 &3
2.1	22/10/11	ML	Example Siri Request Amended

Contents

Traveline API – Guidance for Developers 1
Version control 2
Contents 3
Purpose of the API 4
Architecture 4
Introducing SIRI 5
Introducing NaPTAN 7
Administration 8
Requests and responses 8
Appendix One: SIRI-SM request / response 10
Appendix Two: NextBuses Display Rules 13
Appendix Three: Unused Stops Guidance 14

Purpose of the API

Information on the departure times of buses, coaches, trams and light rail systems in Great Britain is spread across a large number of separate data feeds, with a variety of formats and licensing arrangements. The purpose of the Traveline NextBuses API is to make all this information available as a single standards-based data source.

Departure times feeds fall into two basic categories:

- **Scheduled information**, collected and organised into ten regional feeds by Traveline
- **Real-time information**, typically commissioned on a local basis by local authorities

Traveline currently operates two mobile services for departures information – the SMS service on shortcode 84268 and the NextBuses mobile website. These services connect to a wide and growing range of scheduled and real-time data sources, blending the data and enabling consistent national coverage to be provided (though still with some gaps).

The NextBuses API takes advantage of this integrated set of data sources, making it available under controlled conditions to the developer community, for the purpose of encouraging innovation in creating new services. The API is hosted and operated by Trapeze on behalf of Traveline.

Note that the information provided by the API consists of imminent departure times from a specific stop or station. The user cannot specify future date / time or other parameters, and cannot request a journey plan (i.e. from A to B).

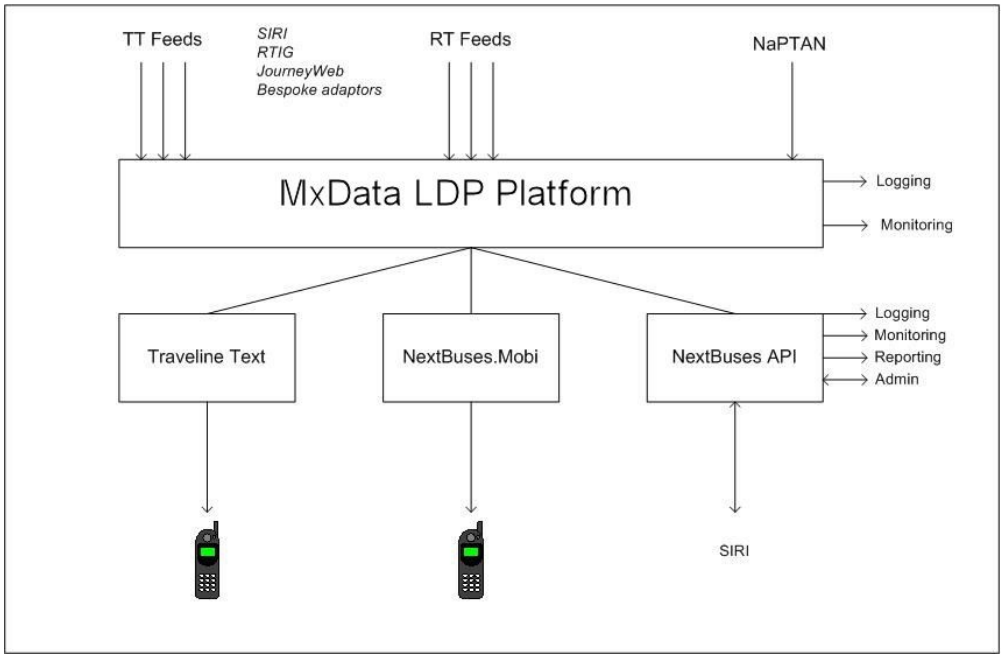
Architecture

The diagram below shows the NextBuses API in the context of the overall set of Traveline departure times services.

The API uses the **SIRI-SM** standard to provide request-response access to the **Trapeze bus feeds infrastructure**, an integration platform that connects to many different local and regional feeds using a variety of formats. **NaPTAN** is used for stop identification.

Business rules are set in this integration layer, including inclusion and exclusion of particular feeds depending on location, naming rules, and determining how information in the feeds is blended when they overlap (typically when there is scheduled and real-time information for the same bus route).

Rules are set by agreement with Traveline, and apply to all services feeding from the integration layer. The data provided by the API will be consistent with that provided to the Traveline SMS and mobile web services



Introducing SIRI

Overview

The **Service Interface for Real-time Information** (SIRI) is a European standard for use in the exchange of information relating to real-time public transport vehicle or journey time data. It is based on the **Transmodel** abstract model for public transport information.

Key elements of the SIRI standard are its communications mechanisms and the message structures for specific functional services. SIRI has an evolving, modular structure allowing users to adopt whichever functions are relevant to their specific needs, and allowing new functional services that use the common protocols to be added in future.

For the purposes of the NextBuses API the relevant communication mechanism is **request / response** and the relevant functional service is **Stop Monitoring** (SIRI-SM). Version 1.0 is currently employed.

The following sections provide background on the overall scope of SIRI. Full technical documentation including examples, schemas and versioning information is available at www.siri.org.uk. Examples of valid request and response SIRI-SM XML are also provided in the appendix below.

SIRI communication mechanisms

The SIRI communications specification makes use of mainstream internet standards, specifically XML over IP networks. There are two basic approaches to SIRI communications; either can be used with any functional service:

- **Publish / subscribe**: server A registers with server B an interest in receiving information of a certain sort, and whenever server B is updated it automatically passes the data to server A

- **Request / response**: if server A wants some information, it sends a request to server B, which is then responded to

SIRI functional services

The functional services relate, in the main, to either timetable or real-time running information, filtered and presented in different ways which are geared to different purposes. Eight functional services are currently defined:

- The **Production Timetable** service provides an up-to-date timetable, including updates. It exchanges the planned schedules of vehicles operating a specific service at a specific time on a particular operational date, giving the vehicle time of arrival and departure at all the timing points on a planned route.
- The **Estimated Timetable** service provides a 'real-time timetable'. It reports progress against the actual schedule of vehicles operating a specific service in real time, detailing expected arrival and departure times at subsequent stops on a planned route. There will be recorded data for stops which have been passed, and predicted data for stops not yet passed. This service allows journeys to be cancelled, added or changed. The service can be used to feed real-time journey planners, and to exchange predicted progress with UTMC systems.
- The **Stop Timetable** service provides a stop-centric view of timetabled vehicle arrivals and departures at a designated stop. It provides an up-to-date stop timetable and is primarily intended as reference data for the Stop Monitoring Service.
- The **Stop Monitoring** service provides real-time information on services due to call at a specific stop. This can be used to provide live departure boards, for example through a web/mobile service provider.
- The **Vehicle Monitoring** service provides information on the current location and status of a specified set of vehicles. It relates to vehicles rather than to specific stops and can be used to support roaming. For instance, a bus operator could use this service to monitor the running of their vehicles which are being tracked across a number of LA areas.
- The **Connection Timetable** service provides information about the scheduled arrivals of vehicles to an interchange point that are designated as making connecting journeys. It is primarily intended as reference data for the Connection Monitoring Service.
- The **Connection Monitoring** service provides updated information in real time about the arrivals of and departures of connecting journeys at an interchange. It is intended to inform passengers on incoming ("feeder") and outgoing ("distributor") vehicles about the status of their connections.
- The **General Message** service is used to exchange general operational information – effectively, it provides an email service within the SIRI system that can transport arbitrary structured and unstructured messages.

Introducing NaPTAN

Overview

The **National Public Transport Access Node** database (NaPTAN) is a UK system for uniquely identifying points of access to public transport.

Every UK railway station, coach terminus, airport, ferry terminal, bus stop, taxi rank or other place where public transport can be joined or left is allocated a unique NaPTAN identifier. There are approximately 450,000 entries in the database.

The relationship of such points to a city, town, village or other locality can be indicated through an association with elements of the **National Public Transport Gazetteer**. The CEN **IFOPT** initiative is underway to develop NaPTAN concepts into a Europe-wide system, as an extension of Transmodel.

For the purposes of the NextBuses API, NaPTAN identifiers must be used to specify the stop for which data is requested using the SIRI-SM request / response communication described above.

The following section provides more information on NaPTAN identifiers. Full technical documentation including schemas, versioning and information on accessing the database is available at http://www.dft.gov.uk/naptan/index.htm.

NaPTAN stop points

NaPTAN identifiers provide systematic identification of all UK points of access to public transport (or **Stop points**, using the Transmodel term). NaPTAN stop points are submitted by transport authorities to a central service which consolidates the data and distributes it back to users. Certain sets of stop point data such as airports and ferry ports are provided nationally.

- Every UK rail station, bus and coach terminus, airport, ferry terminal, bus stop, tram stop, and taxi rank is allocated a unique NaPTAN Identifier
- For large interchanges & termini, NaPTAN points identify the entrances from the public thoroughfare - one identifier is distinguished as the main entrance. Platforms may also be individually identified
- For each stop there are two associated NaPTAN identifiers, each unique within the UK: a 12 character system identifier (the **AtcoCode**), and a short (7 or 8 digit) version suitable for plating on stops and other public facing systems (the **NaptanCode**). This latter number has been designed to be suitable for use in SMS and other delivery channels requiring direct reference to a stop identifier by the general public. It can be keyed easily on a mobile keypad. Either code can be used in the NextBuses API.
- Every local authority has been allocated a unique prefix for their stop numbering, to ensure that stop numbers cannot be duplicated. In England stop details are provided by 87 local authorities and are prefixed with numbers ranging between 010 and 490. In Wales stop details are provided by 22 local authorities and are prefixed with numbers ranging between 511 and 582. In Scotland stop details are provided by 32 local authorities and are prefixed with numbers ranging between 601 and 690.
- In addition there are national number prefixes - 900 for coach stops, 910 for railway stations, 920 for airports, 930 for ferry terminals and 940 for metro and tram stops, these being created centrally and not by local authorities.

Traveline display rules

Traveline has developed a set of display rules to provide a common approach to stop naming using data in NaPTAN. These rules are intended to ensure that a usable public-facing stop name is created given any potential data issues that might exist within the database.

These are set out in Appendix Two and should be followed by all API Developers.

Traveline Unused Stops list

The purpose of the “Unused Stops” database is to enable API Developers to remove such NAPTAN stops from the list of stopping points that they offer to the public This both improves the experience for the end-user and removes unnecessary and unfruitful enquiries being made to the NextBuses system.

Details for developers are set out in Appendix Three.

Administration

Connections to the NextBuses API will be authenticated by username and password (HTTP basic auth) on a server to server basis using credentials provided by Traveline. The login combination will be used to track usage and so it is important that separate logins are set up and used for each service using the API.

Please contact Traveline to request access.

MxData provides the service based on an agreed overall monthly usage limit which will apply across all connections using the service. Please ensure that Traveline is informed of usage expectations so that appropriate infrastructure planning can be undertaken.

Any unexpected high usage that either endangers the service or takes usage materially above the agreed total usage limit for the service will be identified immediately through Trapeze’s automated monitoring systems. MxData will discuss the situation with Traveline, but if urgent action is required a particular connection may be throttled or blocked without notice.

Please contact Traveline with any support issues.

Requests and responses

Request

To use the NextBuses API, POST SIRI-SM v1.0 XML requests to the following URL:

http://<username>:<password>@nextbus.mxdata.co.uk/nextbuses/1.0/1 (production)

Each request must be for a **single stop point**, using the appropriate NaPTAN identifier.

Response

The SIRI-SM XML response will deliver data from relevant feeds for the particular area, and will be in accordance with any business rules set for Traveline NextBuses.

Testing

Test requests can be made with any tool capable of sending HTTP POST data. Examples include HTTP Resource Test (Firefox extension) and cURL (Linux command line tool).

Error responses

In certain cases, the service response will contain error conditions where the service was unable to successfully complete the request.

Any given StopMonitoringRequest may result in an error condition. As with a successful request, the response will contain corresponding StopMonitoringDelivery element, but rather than containing departure information, it will contain error information.

There are a number of different error conditions that can be reported:

- AllowedResourceUsageExceededError – Indicates that the server capacity has been exceeded, or that the 3rd-party data feed is not responding in a timely fashion.
- NoInfoForTopicErrorStructure – Indicates that no bus information can be found for the requested bus stop, either because the stop specified in the request is not recognised, or because no data is available for the stop.
- OtherErrorStructure – A catch-all condition indicating a system error or a failure in the 3rd-party data feed.

Appendix One: SIRI-SM request / response

Example request

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Siri version="1.0" xmlns="http://www.siri.org.uk/">
    <ServiceRequest>
        <RequestTimestamp>2011-10-24T15:09:12Z</RequestTimestamp>
        <RequestorRef>CLIENT_APP_ID</RequestorRef>
        <StopMonitoringRequest version="1.0">
            <RequestTimestamp>2011-10-24T15:09:12Z</RequestTimestamp>
            <MessageIdentifier>12345</MessageIdentifier>
            <MonitoringRef>020035811</MonitoringRef>
        </StopMonitoringRequest>
</ServiceRequest>
</Siri>
```

Key elements

- RequestorRef – identifies the client (username)
- StopMonitoringRequest – container element for a single bus stop request (note that two are included here to show the different types of NaPTAN identifier; but only one should be included in an API request)
- MessageIdentifier – uniquely identifies this particular StopMonitoringRequest
- MonitoringRef – identifier of a single bus stop (ATCO code or the SMS code per NaPTAN)

Example response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Siri version="1.0" xmlns="http://www.siri.org.uk/">
<ServiceDelivery>
<ResponseTimestamp>2011-10-24T15:09:20.889+01:00</ResponseTimestamp>
<StopMonitoringDelivery version="1.0">
<ResponseTimestamp>2011-10-24T15:09:20.889+01:00</ResponseTimestamp>
<RequestMessageRef>12345</RequestMessageRef>
<MonitoredStopVisit>
<RecordedAtTime>2011-10-24T15:09:20.889+01:00</RecordedAtTime>
<MonitoringRef>020035811</MonitoringRef>
<MonitoredVehicleJourney>
<FramedVehicleJourneyRef>
<DataFrameRef>-</DataFrameRef>
<DatedVehicleJourneyRef>-</DatedVehicleJourneyRef>
</FramedVehicleJourneyRef>
<VehicleMode>bus</VehicleMode>
<PublishedLineName>42</PublishedLineName>
<DirectionName>Toddington, The Green</DirectionName>
<OperatorRef>153</OperatorRef>
<MonitoredCall>
<AimedDepartureTime>2011-10-24T15:09:00.000+01:00</AimedDepartureTime>
</MonitoredCall>
</MonitoredVehicleJourney>
</MonitoredStopVisit>
<MonitoredStopVisit>
<RecordedAtTime>2011-10-24T15:09:20.889+01:00</RecordedAtTime>
<MonitoringRef>020035811</MonitoringRef>
<MonitoredVehicleJourney>
<FramedVehicleJourneyRef>
<DataFrameRef>-</DataFrameRef>
```

```
<DatedVehicleJourneyRef>-</DatedVehicleJourneyRef>
</FramedVehicleJourneyRef>
<VehicleMode>bus</VehicleMode>
<PublishedLineName>2</PublishedLineName>
<DirectionName>Elstow, Elstow Park and Ride</DirectionName>
<OperatorRef>5100</OperatorRef>
<MonitoredCall>
<AimedDepartureTime>2011-10-24T15:10:00.000+01:00</AimedDepartureTime>
</MonitoredCall>
</MonitoredVehicleJourney>
</MonitoredStopVisit>
<MonitoredStopVisit>
<RecordedAtTime>2011-10-24T15:09:20.889+01:00</RecordedAtTime>
<MonitoringRef>020035811</MonitoringRef>
<MonitoredVehicleJourney>
<FramedVehicleJourneyRef>
<DataFrameRef>-</DataFrameRef>
<DatedVehicleJourneyRef>-</DatedVehicleJourneyRef>
</FramedVehicleJourneyRef>
<VehicleMode>bus</VehicleMode>
<PublishedLineName>1</PublishedLineName>
<DirectionName>Kempston</DirectionName>
<MonitoredCall>
<AimedDepartureTime>2011-10-24T15:17:00.000+01:00</AimedDepartureTime>
<ExpectedDepartureTime>2011-10-24T15:18:02.000+01:00</ExpectedDepartureTime>
</MonitoredCall>
</MonitoredVehicleJourney>
</MonitoredStopVisit>
<MonitoredStopVisit>
<RecordedAtTime>2011-10-24T15:09:20.889+01:00</RecordedAtTime>
<MonitoringRef>020035811</MonitoringRef>
<MonitoredVehicleJourney>
<FramedVehicleJourneyRef>
<DataFrameRef>-</DataFrameRef>
<DatedVehicleJourneyRef>-</DatedVehicleJourneyRef>
</FramedVehicleJourneyRef>
<VehicleMode>bus</VehicleMode>
<PublishedLineName>53</PublishedLineName>
<DirectionName>Wootton, Squires Road</DirectionName>
<OperatorRef>5100</OperatorRef>
<MonitoredCall>
<AimedDepartureTime>2011-10-24T15:19:00.000+01:00</AimedDepartureTime>
</MonitoredCall>
</MonitoredVehicleJourney>
</MonitoredStopVisit>
<MonitoredStopVisit>
<RecordedAtTime>2011-10-24T15:09:20.889+01:00</RecordedAtTime>
<MonitoringRef>020035811</MonitoringRef>
<MonitoredVehicleJourney>
<FramedVehicleJourneyRef>
<DataFrameRef>-</DataFrameRef>
<DatedVehicleJourneyRef>-</DatedVehicleJourneyRef>
</FramedVehicleJourneyRef>
<VehicleMode>bus</VehicleMode>
<PublishedLineName>2</PublishedLineName>
<DirectionName>Elstow, Elstow Park and Ride</DirectionName>
<OperatorRef>5100</OperatorRef>
<MonitoredCall>
<AimedDepartureTime>2011-10-24T15:22:00.000+01:00</AimedDepartureTime>
</MonitoredCall>
</MonitoredVehicleJourney>
</MonitoredStopVisit>
</StopMonitoringDelivery>
</ServiceDelivery>
</Siri>
```


Key elements

- StopMonitoringDelivery –present for each StopMonitoringRequest in the request
- RequestMessageRef – corresponds to MessageIdentifier in the request
- MonitoringRef – indicates which bus stop this StopMonitoringDelivery relates to

- MonitoredStopVisit – represents a single bus departure
- PublishedLineName –service name
- DirectionName – destination
- AimedDepartureTime –scheduled departure time
- ExpectedDepartureTime –estimated departure time

Appendix Two: NextBuses Display Rules

NextBuses rules for display of real time bus information

Users of NextBuses API on mobile devices are required to present departure time information in the following way

1. The display must show the time at which the request was made, as this is the time from which elapsed times are being calculated
2. Every known departure must be shown whether the times are predicted or not.
3. Scheduled departure times, with no real time predictions, must be shown as 24hr clock time, eg: 13:47
4. Predicted departure times must be shown as a countdown time, eg: 5 mins
5. The scheduled time and the predicted time for a bus must never be presented together (note: this is not the same as the rail industry requirement, where the scheduled time and predicted time are both shown together).
6. The predicted countdown times should be rounded down, so a prediction of 5 mins is valid for 5'59" to 5'00", 4 mins is valid from 4'59" to 4'00" etc
7. A predicted departure time of 1'59" to 1'31" shall be presented as 1 min.
8. A predicted departure time of 1'30" to 0'00" should be presented as "DUE".
9. Scheduled journeys which are not operating should be shown with their scheduled time followed with "CANCELLED".

Users of NextBuses API for display on other channels should follow these principles where possible and should deviate from them only where the context dictates that an alternative approach would be more appropriate.

August 2010

Appendix Three: Unused Stops Guidance

Unused stops data from Ito World

The purpose of the "Unused Stops" database is to enable information service providers to remove such stops from the list of stopping points that they offer to the public – this is particularly relevant for systems using traveline's NextBuses information. This both improves the experience for the end-user and removes unnecessary and unfruitful enquiries being made to the NextBuses system.

Traveline Information Ltd has commissioned *Ito World* to undertake a regular analysis of bus stop and bus schedule data to identify those stops which are shown as ACTive in the NaPTAN database but which are not currently being used by services found in traveline's comprehensive timetable data.

NaPTAN is a database of stops which are *available* to be used – so a bus operator can identify where such approved stops already exist when planning new or revised services. NaPTAN data, therefore, does not immediately react to changes in services and timetables, and it is only by analysing both sets of data regularly that it is possible to identify which NaPTAN stopping points are actually used by services, and which currently are not in use.

Ito reviews both sets of data nationally overnight at least once a week (whenever a new set of timetable data is received from one or more traveline regions) and identifies from the schedule data which NaPTAN stoppoints are referred to in the timetables. It then lists those ACTive stops in the NaPTAN database which are not currently used in the timetable database – these are the "Unused stops".

Traveline's timetable database typically contains details of recent, current and future services. Therefore stops which have recently ceased to be used, or those which will be used by services planned for the future, or those which are used only occasionally, will not be included in the "unused stops" list, even though there may be no services using them "today" or perhaps for many days in the future.