511 Data Exchange including an Open511 Protocol
# Table of Contents

1 Overview ...................................................................................................................................................... 7

2 Transit APIs .................................................................................................................................................. 8

2.1 API: Operator ........................................................................................................................................... 9

2.2 API: Line .................................................................................................................................................. 11

2.3 API: Stop ................................................................................................................................................ 13

2.4 API: StopPlace ...................................................................................................................................... 15

2.5 API: Pattern .......................................................................................................................................... 19

2.6 API: Timetable ....................................................................................................................................... 22

2.7 API: Holidays ......................................................................................................................................... 27

2.8 API: Announcement .............................................................................................................................. 29

2.9 API: Transit Scheduled Departures for a Stop ...................................................................................... 30

2.10 API: Real-time predictions at a Stop ................................................................................................. 31

2.11 API: Real-time Vehicle Monitoring .................................................................................................... 32

2.12 API: Transit Schedule Updates for an agency (Possible Future Implementation) ............................. 33

2.13 API: Transit Addition and Cancellation of Trips by Agency (Possible Future Implementation) ........ 34

2.14 API: General Announcements ........................................................................................................... 35

2.15 API: GTFS-Realtime Trip Updates ...................................................................................................... 35

2.16 API: GTFS-Realtime Vehicle Positions ............................................................................................... 36

2.17 GTFS Operator List .............................................................................................................................. 37

2.18 GTFS DataFeed download ................................................................................................................... 38

2.19 GTFS ServiceAlerts ............................................................................................................................. 40

2.20 API: Shapes .......................................................................................................................................... 40

3 Appendix A: API Response Messages- XML ............................................................................................. 43

3.1 Transit XML .......................................................................................................................................... 43

4 Appendix B: API Response Messages- JSON ............................................................................................ 66

4.1 Transit JSON .......................................................................................................................................... 66

5 Appendix C: API Data Structures ............................................................................................................... 89

5.1 SIRI ......................................................................................................................................................... 89

6 Appendix D: GTFS+ Files Structures ......................................................................................................... 107

7 Appendix E: Historic Regional GTFS Feed ............................................................................................... 109
List of Tables

A.1.1 Example Transit Operator Response (XML) ..................................................43
A.1.2 Example Transit Line Response (XML) .........................................................43
A.1.3 Example Transit Stop Response (XML) .........................................................45
A.1.4 Example Transit Stop Place Response (XML) ..............................................46
A.1.5 Example Transit Pattern Response (XML) ...................................................48
A.1.6 Example Timetable Response (XML) ...........................................................50
A.1.7 Example Transit Holiday Response (XML) ..................................................53
A.1.8 Example Transit Announcement Response (XML) .......................................53
A.1.9 Example Transit Scheduled Departures for a Stop Response (XML) in SIRI ST format ..........................................................54
A.1.10 Example Transit Real Time Predictions at a Stop Response (XML) in SIRI format ..........................................................55
A.1.11 Example Real Time Vehicle Monitoring Response (XML) in SIRI format ..................57
A.1.12 Example Transit Schedule Update Response (XML) in SIRI PT format ..............59
A.1.13 Example Transit Addition and Cancellation of Trip Response (XML) in SIRI ET format .........................................................60
A.1.14 Example Transit General Messaging Service Response (XML) in SIRI GM format ..........................................................61
A.1.15 Example Transit GTFS Operator List in XML format ....................................62
A.1.16 Example Transit ServiceAlerts Response (XML) ........................................62
A.1.17 Example Shapes Response (XML) .............................................................65
B.1.1 Example Transit Operator Response (JSON) ..................................................66
B.1.2 Example Transit Line Response (JSON) ......................................................66
B.1.3 Example Transit Stop Response (JSON) ......................................................67
B.1.4 Example Transit StopPlace Response (JSON) ..............................................68
B.1.5 Example Transit Pattern Response (JSON) ..................................................70
B.1.6 Example Timetable Response (JSON) ..........................................................71
B.1.7 Example Transit Holiday Response (JSON) ..................................................76
B.1.8 Example Transit Announcement Response (JSON) .......................................76
B.1.9 Example Transit Scheduled Departures for a Stop Response (JSON) in SIRI ST format ..........................................................77
B.1.10 Example Transit Real Time Predictions at a Stop Response (JSON) in SIRI format ..........................................................78
B.1.11 Example Real Time Vehicle Monitoring Response (JSON) in SIRI format ...........80
B.1.12 Example Transit Schedule Update Response (JSON) in SIRI PT format ..........82
B.1.13 Example Transit Addition and Cancellation of Trip Response (JSON) in SIRI ET format .........................................................83
B.1.14 Example Transit General Messaging Service Response (JSON) in SIRI GM format ..........................................................84
B.1.15 Example GTFS Operator List in JSON format .............................................85
B.1.16 Example Transit ServiceAlerts Response in JSON format ................................85
B.1.17 Example Transit Shapes Response in JSON format ....................................87
C.1.8 Announcement Message Structure .............................................................89
C.1.9 Transit Scheduled Departures for a Stop Message Structure ..........................90
C.1.10 Real-time predictions at a Stop Message Structure ......................................92
C.1.11 Real-time Vehicle Monitoring Message Structure ......................................98
C.1.12 Transit Schedule Updates for an agency Message Structure ..........................102
C.1.13 Transit Addition and Cancellation of Trips by Agency Message Structure ........104
C.1.14 General Announcements Message Structure ............................................105
C.1.15 ServiceAlerts Structure .................................................................................106
D.1.1 directions.txt File Structure ............................................................................107
D.1.2 calendar_attributes.txt File Structure .........................................................107
D.1.3 farezone_attributes.txt File Structure ........................................................107
D.1.4 rider_categories.txt File Structure ..............................................................108
D.1.5 fare_rider_categories.txt File Structure ........................................................108
E.1.1 Slicing regional feeds .......................................................................................109
## Document History

<table>
<thead>
<tr>
<th>Description</th>
<th>Version</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working Draft - addressed reorganization comments</td>
<td>0.9</td>
<td>08/28/13</td>
</tr>
<tr>
<td>First published version with transit, traffic, tolling, and parking APIs</td>
<td>1.0</td>
<td>09/13/13</td>
</tr>
<tr>
<td>Update Traffic APIs’ structure information, parameters and filters, and their examples to sync with specification provided on Open511.org.</td>
<td>1.0</td>
<td>5/2/2014</td>
</tr>
<tr>
<td>Add GTFS-realtime Trip Updates and Vehicle Positions, and their examples.</td>
<td>1.0</td>
<td>5/7/2014</td>
</tr>
<tr>
<td>Minor updates and corrections</td>
<td>1.0</td>
<td>5/28/2014</td>
</tr>
<tr>
<td>Add sample request endpoint and parameters and filters tables for Section 3.14 and 3.15. Update references for resource endpoints with their exact URL.</td>
<td>1.0</td>
<td>6/12/2014</td>
</tr>
<tr>
<td>Minor updates to Section 3.14 and 3.15</td>
<td>1.0</td>
<td>7/17/2014</td>
</tr>
<tr>
<td>Separated Traffic and Transit</td>
<td>1.0</td>
<td>8/26/2014</td>
</tr>
<tr>
<td>Minor updates to remove references for Traffic</td>
<td>1.0</td>
<td>9/24/2014</td>
</tr>
<tr>
<td>Updated request endpoint URLs for all APIs</td>
<td>1.0</td>
<td>04/06/2016</td>
</tr>
<tr>
<td>Added two new APIs: GTFS Operators List and GTFS Dataset Download. Added sample message response to Section A.1 and B.1</td>
<td>1.0</td>
<td>04/06/2016</td>
</tr>
<tr>
<td>Added missing OperatorRef parameter for Transit Scheduled Departure for a Stop</td>
<td>1.0</td>
<td>04/06/2016</td>
</tr>
<tr>
<td>Marked following two APIs are “Possible Future Implementation”</td>
<td>1.0</td>
<td>04/06/2016</td>
</tr>
<tr>
<td>o Transit Addition and Cancellation of Trips by Agency</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o Transit Schedule Updates for an agency</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Updated JSON output (Section B.1.7) for Holiday API</td>
<td>1.0</td>
<td>04/06/2016</td>
</tr>
<tr>
<td>Added ServiceAlerts API</td>
<td>1.1</td>
<td>06/10/2016</td>
</tr>
<tr>
<td>Updates to Pattern, Timetable and Holiday APIs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o Included Stop Names in Pattern API</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o Added parameter to Timetable API for returning timetables for Special services</td>
<td>1.2</td>
<td>11/08/2016</td>
</tr>
<tr>
<td>o Updated Holiday API to align with GTFS Service Exceptions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Change Description</td>
<td>Version</td>
<td>Date</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------------</td>
<td>---------</td>
<td>------------</td>
</tr>
<tr>
<td>Add Offset parameter in Timetable API</td>
<td>1.21</td>
<td>03/07/2017</td>
</tr>
<tr>
<td>Add element VehicleJourneyName in Stop Timetable API</td>
<td>1.22</td>
<td>05/22/2017</td>
</tr>
<tr>
<td>Updated examples to use the two character Agency ID</td>
<td>1.23</td>
<td>08/14/2017</td>
</tr>
<tr>
<td>Added GTFS+ files in GTFS Data Feed download API and added appendix for GTFS+ files definition</td>
<td>1.24</td>
<td>09/11/2017</td>
</tr>
<tr>
<td>Standardized GTFS Operator API response</td>
<td>1.25</td>
<td>10/02/2017</td>
</tr>
<tr>
<td>Added ExceptionDate parameter to Timetable API Removed daytypes section from Holiday API</td>
<td>1.26</td>
<td>05/30/2018</td>
</tr>
<tr>
<td>Elements added to Stop Monitoring API and Vehicle Monitoring API</td>
<td>1.27</td>
<td>07/01/2019</td>
</tr>
<tr>
<td>Elements and filters added to Stop API. Added Shapes API.</td>
<td>1.28</td>
<td>11/27/2019</td>
</tr>
<tr>
<td>Filters added to GTFS DataFeed download API.</td>
<td>1.29</td>
<td>06/09/2020</td>
</tr>
<tr>
<td>Added Historic Regional GTFS Feed</td>
<td>1.30</td>
<td>06/26/2020</td>
</tr>
</tbody>
</table>
1 Overview

This document focuses on data exchange APIs for the Transit data. For a complete overview of 511 Data Exchange, please refer to Open 511 Data Exchange Specifications – Overview document. The overview document covers:

- General information about 511 Data Exchange
- Different protocols and data feeds available through Open 511 APIs
- Standard Discovery API specifications.
- Encodings and Protocols along with reference to standard documentation.
- Technical Guidelines

It is highly recommended that all users of Open 511 Data Exchange have reviewed the information in the Overview document.
2 Transit APIs

The NeTEx data structures wrapped within the SIRI framework has been adopted for dynamic exchange of Transit service configuration and schedules. Open511 however recommends using HTTP Get method for requests instead of using HTTP Post, as specified by the NeTEx/SIRI standards. The SIRI framework has also been adopted for dynamic exchange of real-time transit data (stop monitoring and vehicle monitoring). GTFS+ standard has been adopted for bulk exchange of static transit configuration data, while GTFS-RT has been adopted for bulk exchange of real-time transit data (service alerts, trip updates and vehicle positions).

The data communication architecture for San Francisco Bay Area 511 is depicted in Figure 1 below.

![Figure 1 – Transit data communication architecture for San Francisco Bay Area 511](image)
All NeTEx responses shall be enclosed within the SIRI ServiceDelivery structure as shown below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Time response was created</td>
</tr>
<tr>
<td>DataObjectDelivery</td>
<td>DataObjects Delivery structure</td>
<td>Mandatory</td>
<td>Delivery for NeTEx service containing one or more NeTEx data objects</td>
</tr>
<tr>
<td>— ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Time individual response element was created</td>
</tr>
<tr>
<td>— dataObjects</td>
<td>Collection of NeTEx dataobjects</td>
<td>Mandatory</td>
<td>NeTEx Entities of any type</td>
</tr>
</tbody>
</table>

### 2.1 API: Operator

Operator within a jurisdiction represents a company providing public transport services. Consumers can request a list of all the operators within the jurisdiction or they can use additional filters such as operator code/id to restrict the results as per their needs and use case.

Below is a message structure of dataObjects for Organisations contained within a NeTEx ResourceFrame. Organisations are a collection of the Operator resource.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceFrame</td>
<td>NeTEx frame</td>
<td>Mandatory</td>
<td>NeTEx container frame for Organizations.</td>
</tr>
<tr>
<td>— organisations</td>
<td>Collection of Operators</td>
<td>Mandatory</td>
<td>A collection of Operator elements. Can contain multiple operator elements, at least one occurrence is mandatory.</td>
</tr>
</tbody>
</table>

**Operator structure**

The operator structure is the main element of the organizations collection. It represents a company providing public transport services.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the operator</td>
</tr>
<tr>
<td>Extensions</td>
<td>Container</td>
<td>Optional</td>
<td>Container for extensions to NeTEx</td>
</tr>
<tr>
<td>—Monitored</td>
<td>Boolean</td>
<td>Optional</td>
<td>Whether agency is real-time enabled or not</td>
</tr>
<tr>
<td>—OtherModes</td>
<td>Enum list</td>
<td>Optional</td>
<td>List of transport modes other than primary mode.</td>
</tr>
<tr>
<td>—Coverage</td>
<td>Container</td>
<td>Optional</td>
<td>Coverage area of the operator – can be a polygon or a list of lines</td>
</tr>
<tr>
<td>—gml:Polygon</td>
<td>GML structure</td>
<td>Optional</td>
<td>GML Polygon representing the coverage</td>
</tr>
<tr>
<td>—gml:LineString</td>
<td>GML structure</td>
<td>Optional</td>
<td>GML Line representing the coverage. Multiple lines can be provided</td>
</tr>
<tr>
<td><strong>PrivateCode</strong></td>
<td>Free Text</td>
<td>Optional</td>
<td>Agency/operator code used within the jurisdiction</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------</td>
<td>----------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td><strong>SiriOperatorRef</strong></td>
<td>Free Text</td>
<td>Optional</td>
<td>An alternative code that uniquely identifies the operator in real-time systems (AVMS)</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the operator.</td>
</tr>
<tr>
<td><strong>ShortName</strong></td>
<td>Free Text</td>
<td>Optional</td>
<td>Short name for the operator.</td>
</tr>
<tr>
<td><strong>Locale</strong></td>
<td>Container</td>
<td>Optional</td>
<td>Container for the operator’s locale information</td>
</tr>
<tr>
<td>— <strong>TimeZone</strong></td>
<td>Free Text</td>
<td>Optional</td>
<td>Timezone Name</td>
</tr>
<tr>
<td>— <strong>DefaultLanguage</strong></td>
<td>Xsd:Language</td>
<td>Optional</td>
<td>Default Language</td>
</tr>
<tr>
<td><strong>ContactDetails</strong></td>
<td>Container</td>
<td>Optional</td>
<td>Container for operator’s contact information</td>
</tr>
<tr>
<td>— <strong>ContactPhoneNumber</strong></td>
<td>Free Text</td>
<td>Optional</td>
<td>Contact telephone number</td>
</tr>
<tr>
<td>— <strong>WebSite</strong></td>
<td>Xsd:AnyURI</td>
<td>Optional</td>
<td>Website address</td>
</tr>
<tr>
<td><strong>PrimaryMode</strong></td>
<td>Enum</td>
<td>Optional</td>
<td>Primary transport mode of operator</td>
</tr>
</tbody>
</table>

**Sample request endpoint for operators**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request EndPoint Example</td>
<td>For e.g. <a href="http://api.511.org/transit/operators?api_key=%7Byour-key%7D">http://api.511.org/transit/operators?api_key={your-key}</a></td>
</tr>
</tbody>
</table>
Parameters and Filters

Parameters and filters supported with the request are shown in the table below. The transit operator response for XML is shown in Appendix A Section A.1.1. The transit operator response for JSON is shown in Appendix B Section B.1.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language, If the jurisdiction doesn’t support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Optional</td>
<td>The operator_id parameter supports filtering based on a particular operator id/code</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual Operator resource cannot be located). For e.g. http://api.511.org/transit/Operators?operator_id=1345&api_key={your-key}&format=json

2.2 API: Line

Lines are routes covered by transit operators within the jurisdiction. Consumers can request list of all the routes within an operator or they can use additional filters like line id to restrict the results as per their needs and use case.

Below is a message structure of dataObjects for lines contained within a NetEx ServiceFrame. Lines are a collection of the Line (Route) resource.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceFrame</td>
<td>NetEx frame</td>
<td>Mandatory</td>
<td>NetEx container frame for Lines.</td>
</tr>
<tr>
<td>—lines</td>
<td>Collection of Lines</td>
<td>Mandatory</td>
<td>A collection of Line elements. Can contain multiple line elements, at least one occurrence is mandatory.</td>
</tr>
</tbody>
</table>
The line structure is the main element of the Lines collection. It represents a route generally known to the public by a name or a number.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the route.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Name of the line.</td>
</tr>
<tr>
<td>TransportMode</td>
<td>Enum</td>
<td>Optional</td>
<td>Mode of transport of line</td>
</tr>
<tr>
<td>PublicCode</td>
<td>Free Text</td>
<td>Optional</td>
<td>Public identifier of the line.</td>
</tr>
<tr>
<td>SiriLineRef</td>
<td>Free Text</td>
<td>Optional</td>
<td>An alternative code that uniquely identifies the operator in real-time systems(AVMS)</td>
</tr>
<tr>
<td>OperatorRef</td>
<td>ID</td>
<td>Mandatory</td>
<td>Reference to the operator for the line</td>
</tr>
<tr>
<td>Monitored</td>
<td>Boolean</td>
<td>Optional</td>
<td>Indicates if real-time data available for line.</td>
</tr>
</tbody>
</table>

Sample request endpoint for lines

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request EndPoint Example</td>
<td>For e.g. <a href="http://api.511.org/transit/lines?api_key=%7Byour-key%7D&amp;operator_id=AC">http://api.511.org/transit/lines?api_key={your-key}&amp;operator_id=AC</a></td>
</tr>
</tbody>
</table>

Parameters and Filters

Parameters and Filters supported with the request are shown in the table below. The transit line response for XML is shown in Appendix A Section A.1.2. The transit line response for JSON is shown in Appendix B Section B.1.2.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language, If the jurisdiction doesn’t support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter limits the search for lines within a particular operator id/code</td>
</tr>
<tr>
<td>Line_id</td>
<td>Optional</td>
<td>The line_id parameter supports filtering based on a particular line</td>
</tr>
</tbody>
</table>
Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual line resource cannot be located). For e.g. http://api.511.org/transit/lines?api_key={your-key}&operator_id=1345

2.3 API: Stop

Stop or ScheduledStopPoint is a location where passengers can board or alight from vehicles. Consumers can request list of all the stops serviced by an agency/operator within the jurisdiction. Stop groupings or StopAreas are also returned when specifically requested using the include_stop_areas parameter.

Below is a message structure of dataObjects for stops contained within a NeTEx ServiceFrame. ScheduledStopPoints are a collection of the ScheduledStopPoint (Stop) resource.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceFrame</td>
<td>NeTEx frame</td>
<td>Mandatory</td>
<td>NeTEx container frame for ScheduledStopPoints.</td>
</tr>
<tr>
<td>— scheduledStopPoints</td>
<td>Collection of ScheduledStop Points</td>
<td>Mandatory</td>
<td>A collection of ScheduledStopPoint elements. Can contain multiple ScheduledStopPoint elements, at least one occurrence is mandatory.</td>
</tr>
<tr>
<td>— stopAreas</td>
<td>Collection of Stop Areas</td>
<td>Optional</td>
<td>A collection of StopArea elements. Stop Areas group stops within an operator. A hierarchy of stop groups could also be provided. The stopAreas are returned only when specifically requested using the include_stop_areas parameter.</td>
</tr>
</tbody>
</table>

ScheduledStopPoint structure

The ScheduledStopPoint structure is the main element of the ScheduledStopPoints collection. It represents a location where passengers can board or alight from vehicles.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the stop.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the stop area.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the stop group (parent stop).</td>
</tr>
<tr>
<td>Members</td>
<td>Container</td>
<td>Optional</td>
<td>Container of stops that belong to the group</td>
</tr>
<tr>
<td>— ScheduledStopPointRef</td>
<td>Reference ID</td>
<td>Optional</td>
<td>ID of the ScheduledStopPoint (within the ‘ref’ attribute)</td>
</tr>
<tr>
<td>ParentStopAreaRef</td>
<td>Reference ID</td>
<td>Optional</td>
<td>Id of the parent stop. Used to build a hierarchy of stop areas. For example, MUNI stops at Embarcadero could be a StopArea 1; ferry stops at Embarcadero could be StopArea 2. Stop Area 3 could be a parent stop area which comprises of all regional transit stops at Embarcadero. Stop Area 3 is then the ParentStopArea for StopArea 1 and 2.</td>
</tr>
</tbody>
</table>

**StopArea structure**

The StopArea structure is the main element of the stopAreas collection. It represents a grouping of stops within or across multiple operators.
Parameters and Filters

Parameters and Filters supported with the request are shown in the table below. The transit stop response for XML is shown in Appendix A Section A.1.3. The transit stop response for JSON is shown in Appendix B Section B.1.3.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language. If the jurisdiction doesn't support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter supports filtering based on a particular operator id/code</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
<tr>
<td>include_stop_areas</td>
<td>Optional</td>
<td>When this parameter is set to true, all stop areas (stop groupings) along with the referenced stops (ScheduledStopPoints) are returned.</td>
</tr>
<tr>
<td>Line_id</td>
<td>Optional</td>
<td>The line_id parameter supports filtering based on a particular route. The line_id should correspond to the id attribute of a Line returned by the Line API.</td>
</tr>
<tr>
<td>Direction_id</td>
<td>Optional</td>
<td>The direction_id parameter supports filtering based on a particular route and direction. This parameter has to be provided along with the line_id parameter. The direction_id should correspond to the id attribute of a Direction returned by the Pattern API for the operator and route.</td>
</tr>
<tr>
<td>Pattern_id</td>
<td>Optional</td>
<td>The pattern_id parameter supports filtering based on a particular pattern. The pattern_id should correspond to the id attribute of a ServiceJourneyPattern returned by the Pattern API.</td>
</tr>
</tbody>
</table>

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual stop resource cannot be located). For e.g. http://api.511.org/transit/stops?api_key={your-key}&operator_id=1345

2.4 API: StopPlace

StopPlace is a named place or the physical stop where public transport may be accessed. Consumers can request list of all the stop places by operator code or they can use additional filters such as stop id to restrict the results as per their needs and use case. For a given stop, the physical representation of the
stop (StopPlace) and the representation of the stop as a point in the timetable (ScheduledStopPoint) will use the same stop identifier (id).

Below is a message structure of dataObjects for lines contained within a NeTEx ServiceFrame. StopPlaces is a collection of the StopPlace resource.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SiteFrame</td>
<td>NeTEx frame</td>
<td>Mandatory</td>
<td>NeTEx container frame for StopPlaces.</td>
</tr>
<tr>
<td>stopPlaces</td>
<td>Collection of StopPlaces</td>
<td>Mandatory</td>
<td>A collection of stopPlace elements. Can contain multiple stopPlace elements, at least one occurrence is mandatory.</td>
</tr>
</tbody>
</table>

**StopPlace structure**

The StopPlace structure is the main element of the stopPlaces collection. It represents a physical stop where public transport may be accessed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the StopPlace.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the StopPlace.</td>
</tr>
<tr>
<td>Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of StopPlace</td>
</tr>
<tr>
<td>Centroid</td>
<td>Container</td>
<td>Optional</td>
<td>Center coordinate of the stopPlace</td>
</tr>
<tr>
<td>Location</td>
<td>Container</td>
<td>Optional</td>
<td>The position of the Point that represents the center of the stopPlace</td>
</tr>
<tr>
<td>—Longitude</td>
<td>Float</td>
<td>Optional</td>
<td>Longitude of stopPlace using WGS84 projection</td>
</tr>
<tr>
<td>—Latitude</td>
<td>Float</td>
<td>Optional</td>
<td>Latitude of stopPlace using WGS84 projection</td>
</tr>
<tr>
<td>AccessibilityAssessment</td>
<td>Container</td>
<td>Optional</td>
<td>The accessibility characteristics of the stopPlace</td>
</tr>
<tr>
<td>—MobilityImpairedAccess</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Summary indication as to whether the stopPlace is considered accessible or not</td>
</tr>
<tr>
<td>—limitations</td>
<td>Container</td>
<td>Optional</td>
<td>Accessibility limitations</td>
</tr>
<tr>
<td>—AccessibilityLimitation</td>
<td>Container</td>
<td>Mandatory</td>
<td>Assessment of the accessibility of the stopPlace</td>
</tr>
<tr>
<td>——WheelChairAccess</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Whether the stopPlace is wheelchair accessible</td>
</tr>
<tr>
<td>alternativeNames</td>
<td>Container</td>
<td>Optional</td>
<td>Container for alternative names</td>
</tr>
<tr>
<td>—AlternativeName</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for Alternative name</td>
</tr>
<tr>
<td>—Name</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Alternative Name</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Optionality</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------</td>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PostalAddress</td>
<td>Container</td>
<td>Optional</td>
<td>Postal address of the stopPlace</td>
</tr>
<tr>
<td>— AddressLine1</td>
<td>Free Text</td>
<td>Optional</td>
<td>First line of address</td>
</tr>
<tr>
<td>— Town</td>
<td>Free Text</td>
<td>Optional</td>
<td>Town</td>
</tr>
<tr>
<td>Url</td>
<td>URI</td>
<td>Optional</td>
<td>Web address of stopPlace</td>
</tr>
<tr>
<td>OperatorRef</td>
<td>Reference ID</td>
<td>Optional</td>
<td>Reference to the operator of the stopPlace (contained in ref attribute of OperatorRef element)</td>
</tr>
<tr>
<td>adjacentSites</td>
<td>Container</td>
<td>Optional</td>
<td>Reference to adjacent sites such as parking locations</td>
</tr>
<tr>
<td>— ParkingRef</td>
<td>Reference ID</td>
<td>Mandatory</td>
<td>Reference to parking associated with the stopPlace (contained in ref attribute of ParkingRef element)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Multiple ParkingRef elements can be included to associate multiple parking locations to the stopPlace</td>
</tr>
<tr>
<td>placeEquipments</td>
<td>Container</td>
<td>Optional</td>
<td>Equipments that may be located in the stopPlace</td>
</tr>
<tr>
<td>— SanitaryEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Container for a sanitary facility such as a restroom, shower, etc.</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of the facility</td>
</tr>
<tr>
<td>— CycleStorageEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Container for cycle storage equipments</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of the facility</td>
</tr>
<tr>
<td>— CycleStorageType</td>
<td>Enum</td>
<td>Optional</td>
<td>Type of storage (e.g. Racks)</td>
</tr>
<tr>
<td>— NumberOfSpaces</td>
<td>Integer</td>
<td>Optional</td>
<td>Number of storage spaces</td>
</tr>
<tr>
<td>— SignEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Sign visible to passengers such as information boards</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of the sign</td>
</tr>
<tr>
<td>— EscalatorEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Escalators in the stopPlace</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of the escalator</td>
</tr>
<tr>
<td>— LiftEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Elevators (Lifts) in the stopPlace</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of the elevator</td>
</tr>
<tr>
<td>— ShelterEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Shelter equipment such as waiting areas</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of shelter</td>
</tr>
<tr>
<td>— SeatingEquipment</td>
<td>Container</td>
<td>Optional</td>
<td>Seating equipment such as benches</td>
</tr>
<tr>
<td>— Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of seating equipment</td>
</tr>
<tr>
<td>PublicCode</td>
<td>Free Text</td>
<td>Optional</td>
<td>Short public code for passengers to use when uniquely identifying the stop</td>
</tr>
<tr>
<td>TransportMode</td>
<td>Enum</td>
<td>Optional</td>
<td>Primary mode of transport associated with the stopPlace</td>
</tr>
<tr>
<td>StopPlaceType</td>
<td>Enum</td>
<td>Optional</td>
<td>Type of stopPlace (for e.g. Rail Station)</td>
</tr>
<tr>
<td>quays</td>
<td>Container</td>
<td>Optional</td>
<td>A collection of quays</td>
</tr>
<tr>
<td>— Quay</td>
<td>Container</td>
<td>Mandatory</td>
<td>A place such as platform where passengers have access to Public transport vehicles</td>
</tr>
<tr>
<td>—— CompassOctant</td>
<td>Enum</td>
<td>Optional</td>
<td>Heading of quay relative to street (E/W/N/S/NE/NW/SE/SW)</td>
</tr>
</tbody>
</table>
### parkings
- **Container**
- **Optional**
  - A collection of parking locations linked to the stopPlace

#### Parking
- **Container**
- **Mandatory**
  - Single parking location

#### Name
- **Free Text**
- **Optional**
  - Name of parking location

#### Description
- **Free Text**
- **Optional**
  - Description

#### Centroid
- **Container**
- **Optional**
  - Container for center location of Parking

#### Location
- **Container**
- **Optional**
  - Center point of Parking

#### Longitude
- **Float**
- **Optional**
  - Longitude of Parking using WGS84 projection

#### Latitude
- **Float**
- **Optional**
  - Latitude of Parking using WGS84 projection

#### PostalAddress
- **Container**
- **Optional**
  - Address of Parking

#### AddressLine1
- **Free Text**
- **Optional**
  - Address Line 1

#### Town
- **Free Text**
- **Optional**
  - Town

#### ParkingType
- **Enum**
- **Optional**
  - Parking type (for e.g. Train station parking, Park and Ride)

#### TotalCapacity
- **Integer**
- **Optional**
  - Total number of parking places

#### RealTimeOccupancyAvailable
- **Boolean**
- **Optional**
  - Whether real time occupancy data available for the parking location

#### parkingAreas
- **Container**
- **Optional**
  - List of Parking areas(Accessible parking, Reserved parking)

#### ParkingArea
- **Container**
- **Mandatory**
  - Parking Area

#### Description
- **Free Text**
- **Optional**
  - Description of area

#### ParkingProperties
- **Container**
- **Optional**
  - Properties of parking area

#### ParkingUserType
- **Enum**
- **Optional**
  - Type of Parking area (for Disabled, Reserved)

#### spaces
- **Container**
- **Optional**
  - Container for parking capacity

#### ParkingCapacity
- **Container**
- **Mandatory**
  - Container for parking capacity

#### NumberOfSpaces
- **Integer**
- **Optional**
  - Number of spaces

#### charges
- **Container**
- **Optional**
  - Parking charges for the parking area

#### tariffBands
- **Container**
- **Optional**
  - Charge bands for parking

#### ParkingTariffChargeBand
- **Container**
- **Mandatory**
  - An area within the parking area for grouping charges (Monthly parking, single day parking, etc.)

#### Description
- **Free Text**
- **Optional**
  - Description of parking charge band

#### MaximumStay
- **Xsd:Duration**
- **Optional**
  - Maximum allowed stay duration for tariff amount

#### Amount
- **Decimal**
- **Optional**
  - Charge for stay

---

**Sample request endpoint for stops**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Endpoint Example</td>
<td>For e.g. <a href="http://api.511.org/transit/stopPlaces?api_key=%7Byour-key%7D&amp;operator_id=AC&amp;stop_id=58538&amp;format=Json">http://api.511.org/transit/stopPlaces?api_key={your-key}&amp;operator_id=AC&amp;stop_id=58538&amp;format=Json</a></td>
</tr>
</tbody>
</table>
Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language. If the jurisdiction doesn't support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter supports filtering based on a particular operator id/code.</td>
</tr>
<tr>
<td>Stop_id</td>
<td>Optional</td>
<td>The stop_id parameter supports filtering based on a particular stop id.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

The transit stop place response for XML is shown in Appendix A Section A.1.4. The transit stop place response for JSON is shown in Appendix B Section B.1.4.

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual StopPlace resource cannot be identified)

2.5 API: Pattern

Pattern is an ordered list of stop points and time points for a Line, it describes a pattern followed by the public transport vehicle. A pattern may pass through the same stoppoint more than once. A Line may consist of more than one pattern.

Below is a message structure of dataObjects for Pattern contained within a NeTEx ServiceFrame.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceFrame</td>
<td>NeTEx frame</td>
<td>Mandatory</td>
<td>NeTEx container frame for directions and journeyPatterns.</td>
</tr>
<tr>
<td>— directions</td>
<td>Collection of Direction</td>
<td>Optional</td>
<td>A collection of Direction elements referenced by the patterns within the journeyPatterns collection. Can contain multiple Direction elements, at least one occurrence is mandatory.</td>
</tr>
</tbody>
</table>
**journeyPatterns**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the Direction.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the Direction.</td>
</tr>
</tbody>
</table>

**Direction structure**

The Direction structure is the main element of the directions collection. It is a classification for the general orientation of a pattern within a Line.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the ServiceJourneyPattern.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the ServiceJourneyPattern.</td>
</tr>
<tr>
<td>Extensions</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for extensions to NeTEx</td>
</tr>
<tr>
<td>— LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Reference to the Line resource.</td>
</tr>
<tr>
<td>Extensions</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for Pattern Headsign</td>
</tr>
<tr>
<td>— FrontText</td>
<td>Free Text</td>
<td>Optional</td>
<td>Pattern Headsign (Should contain Pattern Destination information only)</td>
</tr>
<tr>
<td>pointsInSequence</td>
<td>Container</td>
<td>Mandatory</td>
<td>Contains sequence of points in Servicejourneypattern, points may be scheduledstop points or timingpoints.</td>
</tr>
<tr>
<td>— TimingPointInJourneyPattern</td>
<td>Container</td>
<td>Mandatory</td>
<td>A timing point within the Pattern</td>
</tr>
<tr>
<td>— TimingPointInJourneyPattern</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of TimingPointInJourneyPattern</td>
</tr>
<tr>
<td>TimingPointInJourneyPattern order (attribute)</td>
<td>Positive Integer</td>
<td>Mandatory</td>
<td>Order of Point within PointsInSequence</td>
</tr>
<tr>
<td>ScheduledStopPointRef (attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Identifier of Schedule Stoppoint corresponding to the timing point</td>
</tr>
<tr>
<td>DestinationDisplayView</td>
<td>Container</td>
<td>Optional</td>
<td>If pattern headsign changes at a stop, specify the headsign here</td>
</tr>
<tr>
<td>FrontText</td>
<td>Free Text</td>
<td>Optional</td>
<td>Headsign to display at the stop (Pattern Destination information only)</td>
</tr>
<tr>
<td>Extensions</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for extensions to NeTEx</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Timepoint Stop Name.</td>
</tr>
<tr>
<td>StopPointInJourneyPattern</td>
<td>Container</td>
<td>Mandatory</td>
<td>A stop point within the Pattern</td>
</tr>
<tr>
<td>StopPointInJourneyPattern id (attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of StopPointInJourneyPattern</td>
</tr>
<tr>
<td>StopPointInJourneyPattern order (attribute)</td>
<td>Positive Integer</td>
<td>Mandatory</td>
<td>Order of Point within PointsInSequence</td>
</tr>
<tr>
<td>ScheduledStopPointRef (attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Identifier of Schedule Stoppoint</td>
</tr>
<tr>
<td>DestinationDisplayView</td>
<td>Container</td>
<td>Optional</td>
<td>If pattern headsign changes at a stop, specify the headsign here</td>
</tr>
<tr>
<td>FrontText</td>
<td>Free Text</td>
<td>Optional</td>
<td>Headsign to display at the stop (Pattern Destination information only)</td>
</tr>
<tr>
<td>Extensions</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for extensions to NeTEx</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Stop Name.</td>
</tr>
<tr>
<td>linksInSequence</td>
<td>Container</td>
<td>Optional</td>
<td>Sequence of links (The pattern could be represented as one single link or multiple links in sequence)</td>
</tr>
<tr>
<td>ServiceLinkInJourneyPattern</td>
<td>Container</td>
<td>Optional</td>
<td>ServiceLine in a specified order</td>
</tr>
<tr>
<td>projections</td>
<td>Container</td>
<td>Optional</td>
<td>Projections of the link</td>
</tr>
<tr>
<td>LinkSequenceProjection</td>
<td>Container</td>
<td>Optional</td>
<td>Projection of the link sequence as an ordered series of points</td>
</tr>
<tr>
<td>gml:LineString</td>
<td>Line string</td>
<td>Optional</td>
<td>Series of points representing the link</td>
</tr>
</tbody>
</table>

Sample request endpoint for patterns

<table>
<thead>
<tr>
<th>Request Type</th>
<th>Request Endpoint Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>For e.g. <a href="http://api.511.org/transit/patterns?api_key=%7Byour-key%7D&amp;operator_id=SF&amp;pattern_id=151834">http://api.511.org/transit/patterns?api_key={your-key}&amp;operator_id=SF&amp;pattern_id=151834</a></td>
</tr>
</tbody>
</table>
### Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language. If the jurisdiction doesn’t support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter limits the search for lines within a particular operator id/code</td>
</tr>
<tr>
<td>Pattern_id</td>
<td>Optional</td>
<td>The pattern_id parameter supports filtering based on a particular PatternId</td>
</tr>
<tr>
<td>Line_id</td>
<td>Mandatory</td>
<td>The line_id parameter limits the search for patterns within a particular line id (All patterns for specified line_id will be returned)</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

The transit pattern response for XML is shown in Appendix A Section A.1.5. The transit pattern response for JSON is shown in Appendix B Section B.1.5.

### Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual Journey pattern resource cannot be identified)

### 2.6 API: Timetable

Timetable represents a timetable for a given Line, Direction and DayType. It also contains supporting elements referenced by the timetable such as the Route (ordered list of timepoints for which times are provided), day type (service type) and optionally day assignments (assignment of a daytype to each day within the service period). When the IncludeSpecialService parameter is set to true, this API returns all the holiday services (if any) for specified Line. When the IncludeSpecialService parameter is set to false or the IncludeSpecialService parameter is not provided, it returns all the regular timetables for the specified Line. When the ExceptionDate parameter is set to a service exception date (one of the dates returned by the Holiday API for the same agency), the API returns the exception/holiday timetable for the specified line and date. If no timetables are returned, it shall be assumed that the agency is not providing any service for the line on the given exception date.

Below is a message structure of dataObjects for Timetable within a NeTEx CompositeFrame.
### CompositeFrame
- **Type**: NeTEx frame
- **Mandatory**: Mandatory
- **Description**: NeTEx container version Frame that groups a set of content version frames to which same validity conditions have been assigned.

### ServiceFrame
- **Type**: NeTEx frame
- **Mandatory**: Mandatory
- **Description**: NeTEx container frame for routes which is collection of Route. Route represents an ordered list of timepoint stops for which times are provided in the timetable. Multiple routes could be provided in cases where multiple timetables are returned. Each timetable would reference the appropriate route for the timetable.

### ServiceCalendarFrame
- **Type**: NeTEx frame
- **Mandatory**: Mandatory
- **Description**: NeTEx container frame for collection of DayType and DayTypeAssignments. Should contain at least one DayType. DayTypeAssignments are returned only if requested specifically using the input parameter(flag) IncludeDayTypeAssignments

### TimetableFrame
- **Type**: NeTEx frame
- **Mandatory**: Mandatory
- **Description**: NeTEx container frame for a timetable. Multiple TimetableFrames can be returned, one per timetable. The id attribute of the TimetableFrame should be unique across all timetables.

### Service Calendar Frame Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceFrame</td>
<td>NeTEx frame</td>
<td>Mandatory</td>
<td>NeTEx container frame for routes.</td>
</tr>
<tr>
<td>— routes</td>
<td>Collection of Routes</td>
<td>Mandatory</td>
<td>A collection of Route elements. Can contain multiple Route elements, at least one occurrence is mandatory.</td>
</tr>
</tbody>
</table>

### Route Structure

The Route structure is the main element of the routes collection. At least one Route is mandatory within the routes. Route represents an ordered list of timepoint stops for which times are provided in the timetable.
**id (Attribute)** | Free Text | **Mandatory** | Unique identifier of the Route.
---|---|---|---
Name | Free Text | **Optional** | Name for the Route.
LineRef | ID | **Mandatory** | Reference to the Line, ref attribute contains identifier of the line
DirectionRef | ID | **Mandatory** | Reference to the Direction, ref attribute contains identifier to the Direction
pointsInSequence | Container | **Mandatory** | Container for ordered set of time points making up the Route. It should contain at least 2 PointOnRoute
— PointOnRoute | Free Text | **Mandatory** | It is the reference to the ordered route points of Route, id attribute contains unique identifier for PointOnRoute
—— PointRef | ID | **Mandatory** | It is reference scheduled stoppoint representing the timepoint, ref attribute contains identifier to the point

**DayType Structure**

The dayTypes structure contains the collection of DayTypes referenced by the timetables. DayType is a type of day characterized by one or more properties which affect public transport operation.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th><strong>Mandatory/Optional</strong></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td><strong>Mandatory</strong></td>
<td>Unique identifier of the DayType.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td><strong>Mandatory</strong></td>
<td>Name of the DayType.</td>
</tr>
<tr>
<td>properties</td>
<td>Container</td>
<td><strong>Mandatory</strong></td>
<td>Container for the list of PropertyOfDay. Should contain at least one PropertyOfDay.</td>
</tr>
<tr>
<td>— PropertyOfDay</td>
<td>Container</td>
<td><strong>Mandatory</strong></td>
<td>A container for DaysOfWeek property.</td>
</tr>
<tr>
<td>—— PropertyOfDayGroup</td>
<td>Enum</td>
<td><strong>Mandatory</strong></td>
<td>It contains DaysOfWeek logically appended together</td>
</tr>
</tbody>
</table>

**DayTypeAssignment structure**

The dayTypeAssignments structure contains the collection of DayTypeAssignments, which links every operating day within the service period to a daytype. The service period is defined within the Timetable Frame.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th><strong>Mandatory/Optional</strong></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td><strong>Mandatory</strong></td>
<td>Unique identifier of the DayTypeAssignment.</td>
</tr>
</tbody>
</table>
### TimetableFrame structure

TimetableFrame is a coherent set of timetable data which consist of vehicle Journeys and blocks to which the same validity condition has been assigned.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the TimetableFrame.</td>
</tr>
<tr>
<td>Name</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the TimetableFrame.</td>
</tr>
<tr>
<td>frameValidityConditions</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for the AvailabilityCondition which applies to whole Timetable.</td>
</tr>
<tr>
<td>— AvailabilityCondition</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is a container for available conditions.</td>
</tr>
<tr>
<td>—— FromDate</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Start date of Timetable validity period.</td>
</tr>
<tr>
<td>—— ToDate</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>End date of Timetable validity period.</td>
</tr>
<tr>
<td>—— dayTypes</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for DayType reference. It lists the daytypes referenced by the timetable.</td>
</tr>
<tr>
<td>—— DayTypeRef</td>
<td>ID</td>
<td>Mandatory</td>
<td>It is a reference to DayType, ref attribute has reference value to a DayType</td>
</tr>
<tr>
<td>vehicleJourneys</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for collection of ServiceJourney (Trip).</td>
</tr>
<tr>
<td>— ServiceJourney</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>ServiceJourney is a planned movement of public transport on a DayType. Id attribute has unique identifier for Service Journey</td>
</tr>
<tr>
<td>—— SiriVehicleJourneyRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>An alternative code that uniquely identifies the journey. Specifically for use in AVMS systems</td>
</tr>
<tr>
<td>—— JourneyPatternView</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is a container for simplified journey pattern view</td>
</tr>
<tr>
<td>——— ServiceJourneyPatternRef</td>
<td>ID</td>
<td>Mandatory</td>
<td>Reference to Service Pattern, ref attribute contains identifier for service journey Pattern</td>
</tr>
<tr>
<td>——— RouteRef</td>
<td>ID</td>
<td>Mandatory</td>
<td>Reference to Route, ref attribute contains identifier for Route</td>
</tr>
<tr>
<td>——— DirectionRef</td>
<td>ID</td>
<td>Mandatory</td>
<td>Reference to Direction, ref attribute contain identifier for Direction</td>
</tr>
<tr>
<td>— calls</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is container for complete sequence of stops along the route path.</td>
</tr>
<tr>
<td>— call</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is a visit to a scheduled stop point as part of a vehicle journey, order attribute</td>
</tr>
</tbody>
</table>
Sample request endpoint for timetable

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Endpoint Example</td>
<td>For e.g. <a href="http://api.511.org/transit/timetable?api_key=%7Byour-key%7D&amp;operator_id=BA&amp;line_id=COLS/OAKL">http://api.511.org/transit/timetable?api_key={your-key}&amp;operator_id=BA&amp;line_id=COLS/OAKL</a></td>
</tr>
</tbody>
</table>

Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language. If the jurisdiction doesn’t support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter supports filtering based on a particular operator id/code</td>
</tr>
<tr>
<td>Line_id</td>
<td>Mandatory</td>
<td>The line_id parameter supports filtering based on a particular line id. All timetables for the line are returned</td>
</tr>
<tr>
<td>IncludeDayTypeAssignments</td>
<td>Optional</td>
<td>DayTypeAssignments will be included only if this flag is set to true.</td>
</tr>
</tbody>
</table>
api_key | Mandatory | Unique key assigned to a user after they signup for Open511.

IncludeSpecialService | Optional | The timetables for service exceptions for the selected Line are returned (if available) when this parameter is set to ‘true’. When this parameter is omitted or the value is set to ‘false’, service exceptions are not included. Any value other than ‘true’ or ‘false’ will result in a 404 error.

ExceptionDate | Optional | When this parameter is set to one of the dates returned by the Holiday API for the same agency, the exception/holiday timetable for the given line and the exception/holiday is returned. When no timetables are returned for an exception date (no TimetableFrame elements), it should be assumed that the agency is not providing any service for the line on the date. The ExceptionDate should be provided in the yyyyMMdd format.

The transit timetable response for XML is shown in Appendix A Section A.1.6. The transit timetable response for JSON is shown in Appendix B Section B.1.6.

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual Holiday resource cannot be identified)

2.7 API: Holidays

Holidays is a collection of service exceptions defined by an agency or operator.

Below is a message structure of dataObjects for service exceptions contained within a NeTEx ServiceCalendarFrame.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceCalendarFrame</td>
<td>NeTEx frame</td>
<td>Mandatory</td>
<td>NeTEx container frame for service exceptions and dayTypes.</td>
</tr>
<tr>
<td>— ServiceCalendar</td>
<td>ServiceCalendar</td>
<td>Mandatory</td>
<td>Represents the service period.</td>
</tr>
</tbody>
</table>
### contentValidityConditions

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the ServiceCalendar.</td>
</tr>
<tr>
<td>FromDate</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Start date of ServiceCalendar (Service Period)</td>
</tr>
<tr>
<td>ToDate</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>End date of ServiceCalendar (Service Period)</td>
</tr>
</tbody>
</table>

### ServiceCalendar structure

The ServiceCalendar structure represents the service period for the service exceptions.

### AvailabilityCondition structure

The AvailabilityCondition structure is the main element of the contentValidityConditions collection. Every exception/holiday date defined by the agency is provided as an AvailabilityCondition element.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id (Attribute)</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier of the AvailabilityCondition.</td>
</tr>
<tr>
<td>Description</td>
<td>Free Text</td>
<td>Optional</td>
<td>Description of the AvailabilityCondition.</td>
</tr>
<tr>
<td>FromDate</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Start date of AvailabilityCondition(Service Exception)</td>
</tr>
<tr>
<td>ToDate</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>End date of AvailabilityCondition(Service Exception)</td>
</tr>
</tbody>
</table>

### Sample request endpoint for stops

**Request Type**

GET

For e.g.

http://api.511.org/transit/holidays?api_key={your-key}&operator_id=SF

**Parameters and Filters supported with the request**
## Parameter | Mandatory/Optional | Description
--- | --- | ---
Format | Optional | The response format (json/xml) desired. If none specified, then default response would be JSON.
accept_language | Optional | If multiple languages are supported, this can be used to request data in desired language. If the jurisdiction doesn’t support the response in requested language, response could be in default language selected by jurisdiction.
Operator_id | Mandatory | The operator_id parameter supports filtering based on a particular operator id/code.
api_key | Mandatory | Unique key assigned to a user after they signup for Open511.

The transit Holidays response for XML is shown in Appendix A Section A.1.7. The transit Holidays response for JSON is shown in Appendix B Section B.1.7.

### Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- **500** - Internal Server Error (System has issues processing your request)
- **401** – Unauthorized (Invalid API key)
- **404** – Not found (If an individual Holiday resource cannot be identified)

### 2.8 API: Announcement

Announcement is completely SIRI entity; it is a description of a situation/condition about the public transport. Announcement consists of Situations which is collection of PtSituationElement which contains description of situation/condition, at least one PtSituationElement is mandatory.

A message structure of PtSituationElement for Announcement contained within Situations is shown in Appendix C Section C.1.8.

### Sample request endpoint for stops

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>For e.g.</td>
<td><a href="http://api.511.org/transit/transitannouncements?api_key=%7Byour-key%7D">http://api.511.org/transit/transitannouncements?api_key={your-key}</a></td>
</tr>
</tbody>
</table>

### Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
</tbody>
</table>
Operator_id | Optional | The operator_id parameter supports filtering based on a particular operator id/code
--- | --- | ---
Line_id | Optional | The line_id parameter supports filtering based on a particular line id
api_key | Mandatory | Unique key assigned to a user after they signup for Open511.

The transit announcement response for XML is shown in Appendix A Section A.1.8. The transit announcement response for JSON is shown in Appendix B Section B.1.8.

**Possible Errors**

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual Announcement resource cannot be identified)

2.9 API: Transit Scheduled Departures for a Stop

SIRI Stop Timetable service provides static/scheduled timetables in the system for a particular stop. A message structure of Transit Scheduled Departures in SIRI ST (Stop Timetable) format which consists of a single ServiceDelivery node containing details on scheduled visits to this stop within a departure window is shown in Appendix C Section C.1.9.

**Sample request endpoint**

| Request Type | GET | For e.g. http://api.511.org/transit/stoptimetable?api_key={your-key}&MonitoringRef=13008&OperatorRef=SF |

**Parameters and Filters supported with the request**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>LineRef</td>
<td>Optional</td>
<td>The RouteCode that uniquely identifies a transit route.</td>
</tr>
<tr>
<td>OperatorRef</td>
<td>Mandatory</td>
<td>The operator_id parameter supports filtering based on a particular operator id/code</td>
</tr>
<tr>
<td>MonitoringRef</td>
<td>Mandatory</td>
<td>The StopCode that uniquely identifies a physical stop or platform.</td>
</tr>
</tbody>
</table>
### StartTime

Optional

The start date parameter allows for requesting departures within a departure window.

### EndTime

Optional

The end date parameter allows for requesting departures within a departure window.

### api_key

Mandatory

Unique key assigned to a user after they signup for Open511.

The example response for XML in SIRI ST format is shown in Appendix A Section A.1.9. The example response for JSON in SIRI ST format is shown in Appendix B Section B.1.9.

#### 2.10 API: Real-time predictions at a Stop

Siri Stop Monitoring service provides current and forthcoming vehicles arrivals and departures at a stop.

A message structure of real-time departures which consists of a single **ServiceDelivery** node containing details on monitored visits to this stop is shown in Appendix C Section C.1.10

**Sample request endpoint**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>For e.g. <a href="http://api.511.org/transit/StopMonitoring?api_key=%7Byour-key%7D">http://api.511.org/transit/StopMonitoring?api_key={your-key}</a> &amp;agency=AC</td>
</tr>
</tbody>
</table>

**Parameters and Filters supported with the request**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
<tr>
<td>agency</td>
<td>Mandatory</td>
<td>Agency ID to be monitored (e.g. actransit)</td>
</tr>
<tr>
<td>stopCode</td>
<td>Optional</td>
<td>Numeric stop code for the stop to be monitored. When stop code is not provided, the API will return all available information for all stops. Depending on the amount of data, the response time for the API can be more than 5-7 seconds.</td>
</tr>
</tbody>
</table>

The transit real time departure service delivery mode response for XML is shown in Appendix A Section A.1.10 and for JSON is shown in Appendix B Section B.1.10.
Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If a resource cannot be located)

2.11 API: Real-time Vehicle Monitoring

Siri Vehicle monitoring service provides information about current location and expected activities of a particular vehicle. It also provides details for current and subsequent journey patterns. A message structure for real-time vehicle/trip monitoring which consists of a single ServiceDelivery node containing details on vehicle/trip within an agency that are currently operational and being monitored is shown in Appendix C Section C.1.1.

Sample request endpoint

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Endpoint Example</td>
<td>For e.g. <a href="http://api.511.org/transit/VehicleMonitoring?api_key=%7Byour-key%7D&amp;agency=AC">http://api.511.org/transit/VehicleMonitoring?api_key={your-key}&amp;agency=AC</a></td>
</tr>
</tbody>
</table>

Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
<tr>
<td>agency</td>
<td>Mandatory</td>
<td>Agency ID to be monitored (e.g. AC)</td>
</tr>
<tr>
<td>vehicleID</td>
<td>Optional</td>
<td>The unique identifier of the vehicle to be monitored.</td>
</tr>
</tbody>
</table>

The real time vehicle monitoring response for XML in SIRI format is shown in Appendix A Section A.1.11 and for JSON is shown in Appendix B Section B.1.11.

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
• 404 – Not found (If a resource cannot be located)

2.12 API: Transit Schedule Updates for an agency (Possible Future Implementation)

Siri Production Timetable provides information about the expected operation of a transport network for a specified day.

A message structure of Transit Schedule Updates in SIRI PT (Production Timetable) format which consists of a single ServiceDelivery node containing details on schedule updates for a specific line and direction by an agency is shown in Appendix C Section C.1.12.

Sample request endpoint

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Endpoint Example</td>
<td></td>
</tr>
</tbody>
</table>

Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>OperatorRef</td>
<td>Mandatory</td>
<td>The Agency Name that uniquely identifies a transit agency.</td>
</tr>
<tr>
<td>Lineref</td>
<td>Optional</td>
<td>The unique identifier or a transit route. Value could either be RouteCode or RouteName as required. Recommend RouteCode because response has &quot;PublishedLineName&quot; as RouteName.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Optional</td>
<td>Direction (ID) for the route.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

The transit schedule update response for XML is shown in Appendix A Section A.1.12. The transit schedule update response for JSON is shown in Appendix B Section B.1.12.

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

• 500 - Internal Server Error (System has issues processing your request)
2.13 API: Transit Addition and Cancellation of Trips by Agency (Possible Future Implementation)

Siri Estimated Timetable service provides details of the operation of the transport network for a period within the current day, detailing real time deviations from the timetables and control actions affecting the Timetable (cancellations, additional Journeys and Detours).

A message structure of Transit Addition and Cancellation of Trips in SIRI ET (Estimated Timetable) format which consists of a single ServiceDelivery node containing details on schedule updates for a specific line and direction by an agency is shown in Appendix C Section C.1.13.

Sample request endpoint

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Endpoint Example</td>
<td></td>
</tr>
</tbody>
</table>

Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>OperatorRef</td>
<td>Mandatory</td>
<td>The Agency Name that uniquely identifies a transit agency.</td>
</tr>
<tr>
<td>Lineref</td>
<td>Optional</td>
<td>The unique identifier or a transit route. Value could either be RouteCode or RouteName as required. Recommend RouteCode because response has &quot;PublishedLineName&quot; as RouteName.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Optional</td>
<td>Direction (ID) for the route.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

The transit addition and cancellation response for XML is shown in Appendix A Section A.1.13 and for JSON is shown in Appendix B Section B.1.13.

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:
2.14 API: General Announcements

Siri General Messaging Service provides a structured way to exchange arbitrary informative messages between participants, such as travel news, or operational advice.

A message structure of Service Announcements in SIRI GM (General Message) format which consists of a single ServiceDelivery node containing details on general messages is shown in Appendix C Section C.1.14.

**Sample request endpoint**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>For e.g. <a href="http://api.511.org/transit/GeneralAnnouncements?api_key=%7Byour-key%7D">http://api.511.org/transit/GeneralAnnouncements?api_key={your-key}</a></td>
</tr>
</tbody>
</table>

**Parameters and Filters supported with the request**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

The transit general messaging service response for XML is shown in Appendix A Section A.1.14 and for JSON is shown in Appendix B Section B.1.14.

**Possible Errors**

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If a resource cannot be located)

2.15 API: GTFS-Realtime Trip Updates

GTFS-realtime trip updates service response format type is based on Protocol Buffers, Section B.1.

**Sample request endpoint**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>For e.g. <a href="http://api.511.org/Transit/TripUpdates?api_key=%7Byour-key%7D&amp;agency=AC">http://api.511.org/Transit/TripUpdates?api_key={your-key}&amp;agency=AC</a></td>
<td></td>
</tr>
</tbody>
</table>

**Parameters and Filters supported with the request**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
</table>

- **format** *
  - Conditional: mandatory if `Accept: application/x-google-protobuf` (or) `Accept: application/octet-stream` is not provided in HTTP header.
  - The response format protobuf desired. If none specified, then `Accept: application/x-google-protobuf` (or) `Accept: application/octet-stream` must be provided in HTTP header.

- **agency** *
  - Mandatory
  - Agency ID to be monitored (e.g. AC)

- **api_key** *
  - Mandatory
  - Unique key assigned to a user after they signup for Open511.

**Possible Errors**

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If a resource cannot be located)

2.16 API: GTFS-Realtime Vehicle Positions


GTFS-realtime vehicle position service response format type is based on Protocol Buffers, Section B.2.

**Sample request endpoint**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
</table>

June 26, 2020
Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Conditional</td>
<td>Conditional: mandatory if Accept: application/x-google-protobuf (or) Accept: application/octet-stream is not provided in HTTP header. The response format protobuf desired. If none specified, then Accept: application/x-google-protobuf (or) Accept: application/octet-stream must be provided in HTTP header.</td>
</tr>
<tr>
<td>agency</td>
<td>Mandatory</td>
<td>Agency ID to be monitored (e.g. AC)</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If a resource cannot be located)

2.17 GTFS Operator List

GTFS Operator List is the list of operators/agencies that have GTFS dataset available via Open511 APIs. This API also lists the operator that represents the regional GTFS feed.

Sample request endpoint

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>For e.g. <a href="http://api.511.org/transit/gtfsoperators?api_key=%7Byour-key%7D">http://api.511.org/transit/gtfsoperators?api_key={your-key}</a></td>
</tr>
</tbody>
</table>

Below is a message structure of GTFSAgenciesList which is the main element of XML response for this API.
### Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTFSAgency</td>
<td>XML Element – Container</td>
<td>Mandatory</td>
<td>Parent element for each operator/agency providing details about that agency/operator</td>
</tr>
<tr>
<td>—Id</td>
<td>XML Attribute – Text</td>
<td>Mandatory</td>
<td>XML Attribute text value providing Carrier ID (Operator/Agency ID). Note: ID ‘RG’ represents regional GTFS feed</td>
</tr>
<tr>
<td>—Name</td>
<td>XML Attribute – Text</td>
<td>Mandatory</td>
<td>XML Attribute text value providing Carrier Name (Operator/Agency Name). Note: Name value ‘Regional GTFS’ represents regional GTFS Feed</td>
</tr>
<tr>
<td>— LastGenerated</td>
<td>XML Attribute – Text</td>
<td>Mandatory</td>
<td>XML Attribute text value providing timestamp when the last GTFS dataset was generated for this operator. The timestamp is in following format: MM/dd/yyyy HH:mm:ss [AM</td>
</tr>
</tbody>
</table>

**Parameters and Filters supported with the request**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>

The transit GTFS Operator response for XML is shown in Appendix A Section A.1.15. The transit GTFS response for JSON is shown in Appendix B Section B.1.15.

**Possible Errors**

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual resource cannot be identified)

### 2.18 GTFS DataFeed download

GTFS datafeed download allows the user to download a zip file containing GTFS dataset for the specified operator/agency or the regional feed.

The zip file contains the text files corresponding to the GTFS file formats. It also contains additional files, called the GTFS+ files, that provide information that is not contained in the GTFS files such as the
direction names, farezone names, etc. The list of GTFS+ files and their data structures are provided in Appendix D of this document.

When the request is processed successfully, the user will receive a zip file attachment in response to this API.

The regional GTFS feed can be downloaded by setting the operator_id parameter to the value ‘RG’.

Historic regional GTFS feeds can be downloaded by using the historic parameter when the operator_id parameter is set to ‘RG’. Additional information about Historic Regional GTFS Feed is available in Appendix E.

**Sample request endpoint**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>For e.g.</td>
<td><a href="http://api.511.org/transit/datafeeds?api_key=%7Byour-key%7D&amp;operator_id=BG">http://api.511.org/transit/datafeeds?api_key={your-key}&amp;operator_id=BG</a></td>
</tr>
</tbody>
</table>

**Parameters and Filters supported with the request**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter supports filtering based on a particular operator id/code. These operator codes/IDs can be retrieved from CarrierID filed in the GTFS Operator List API response. The operator_id ‘RG’ can be used to download regional GTFS dataset.</td>
</tr>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
</tbody>
</table>
| historic  | Optional           | Use this parameter to download historic regional feeds. This parameter should be used only with operator_id=RG. The value should be set as below:
- A single month should be denoted with year and month in the format YYYY-MM (For example, ‘2020-02’ for February 2020)

**Possible Errors**

Listed below are HTTP status code and message returned for certain common errors:

- **500 - Internal Server Error** (System has issues processing your request)
- **401 – Unauthorized** (Invalid API key)
2.19 GTFS ServiceAlerts
A GTFS dataset for Service Alerts. Service Alerts allow you to provide updates whenever there is disruption on the network.

Data formats supported are: JSON, XML, and Protobuf (default).

Sample request endpoint

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://api.511.org/transit/servicealerts?api_key=%7Byour-key%7D">http://api.511.org/transit/servicealerts?api_key={your-key}</a></td>
<td></td>
</tr>
</tbody>
</table>

Parameters and Filters supported with the request

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>api_key</td>
<td>Mandatory</td>
<td>Unique key assigned to a user after they signup for Open511.</td>
</tr>
<tr>
<td>format</td>
<td>Optional</td>
<td>“json” to receive a JSON response or “xml” to receive an XML response</td>
</tr>
<tr>
<td>agency</td>
<td>Optional</td>
<td>When Agency/Operator ID are provided, the service alerts are filtered by the agency ID. These IDs could be obtained from operators API endpoint.</td>
</tr>
</tbody>
</table>

Possible Errors

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual resource cannot be identified)

2.20 API: Shapes
Shapes describe the path that a vehicle travels along a trip. Consumers can request the shape for an agency's trip id. Trip ids can be obtained using the Timetable API.

Below is a message structure of dataObjects for shapes contained within a NeTEx TimetableFrame.
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vehicleJourneys</td>
<td>Container</td>
<td>Mandatory</td>
<td>Container for collection of ServiceJourney (Trip).</td>
</tr>
<tr>
<td>—ServiceJourney</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>ServiceJourney is a planned movement of public transport, which is equivalent to a trip. Id attribute has unique identifier for the trip.</td>
</tr>
<tr>
<td>———LinkSequenceProjection</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Corresponds to a shape, which is a collection of ordered points along the path of the vehicle. Id attribute has unique identifier for the shape.</td>
</tr>
<tr>
<td>———gml:LineString</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Special curve that consists of a single segment with linear interpolation. The ‘srsName’ attribute specifies the coordinate reference system used.</td>
</tr>
<tr>
<td>————gml:pos</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Coordinates for a point within the line, per the coordinate reference system specified in the srsName attribute of <a href="">gml:LineString</a>.</td>
</tr>
</tbody>
</table>

**Sample request endpoint for lines**

<table>
<thead>
<tr>
<th>Request Type</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request EndPoint Example</td>
<td>For e.g. <a href="http://api.511.org/transit/shapes?api_key=%7Byour-key%7D&amp;operator_id=BA">http://api.511.org/transit/shapes?api_key={your-key}&amp;operator_id=BA</a> &amp;trip_id=3010811SUN</td>
</tr>
</tbody>
</table>

**Parameters and Filters**

Parameters and Filters supported with the request are shown in the table below. The shape response for XML is shown in Appendix A Section A.1.18. The transit line response for JSON is shown in Appendix B Section B.1.18.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Optional</td>
<td>The response format (json/xml) desired. If none specified, then default response would be JSON.</td>
</tr>
<tr>
<td>accept_language</td>
<td>Optional</td>
<td>If multiple languages are supported, this can be used to request data in desired language. If the jurisdiction doesn’t support the response in requested language, response could be in default language selected by jurisdiction.</td>
</tr>
<tr>
<td>Operator_id</td>
<td>Mandatory</td>
<td>The operator_id parameter specifies the operator id/code for which the shape is requested.</td>
</tr>
</tbody>
</table>
### trip_id

**Mandatory**
The trip_id parameter specifies the trip id for which the shape is requested.

### api_key

**Mandatory**
Unique key assigned to a user after they signup for Open511.

**Possible Errors**

Listed below are HTTP status code and message returned for certain common errors:

- 500 - Internal Server Error (System has issues processing your request)
- 401 – Unauthorized (Invalid API key)
- 404 – Not found (If an individual shape resource cannot be located). For e.g. http://api.511.org/transit/shapes?api_key={your-key}&operator_id=1345&trip_id=123
Appendix A: API Response Messages - XML

3.1 Transit XML

A.1.1 Example Transit Operator Response (XML)

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
  <siri:ServiceDelivery>
    <siri:ResponseTimestamp>2012-12-17T09:30:46-05:00</siri:ResponseTimestamp>
    <siri:ResponseTimestamp>2012-12-17T09:30:47.0Z</siri:ResponseTimestamp>
    <DataObjectDelivery>
      <ResourceFrame id="RF" version="any">
        <organisations>
          <Operator id="SF" version="any">
            <Extensions>
              <Monitored>true</Monitored>
              <OtherModes>tram funicular</OtherModes>
              <Coverage>
                <gml:Polygon srsName="EPSG:4326">
                  <gml:coordinates>-71.17,47.33 -71.15,47.36 -71.10,47.35 -71.20,47.40</gml:coordinates>
                </gml:Polygon>
              </Coverage>
            </Extensions>
            <PrivateCode>SF</PrivateCode>
            <SiriOperatorRef>SF</SiriOperatorRef>
            <Name>Muni (San Francisco)</Name>
            <ShortName>Muni</ShortName>
            <Locale>
              <TimeZone>America/Vancouver</TimeZone>
              <DefaultLanguage>en</DefaultLanguage>
            </Locale>
            <ContactDetails>
              <ContactTelephoneNumber>1-415-701-2311</ContactTelephoneNumber>
              <WebSite>http://www.sfmta.com/</WebSite>
            </ContactDetails>
            <PrimaryMode>bus</PrimaryMode>
          </Operator>
        </organisations>
      </ResourceFrame>
    </DataObjectDelivery>
  </siri:ServiceDelivery>
</siri:Siri>
```

A.1.2 Example Transit Line Response (XML)
<?xml version="1.0" encoding="iso-8859-1"?>
    <siri:ServiceDelivery>
        <siri:ResponseTimestamp>2013-09-09T16:55:24-08:00</siri:ResponseTimestamp>
        <DataObjectDelivery>
            <siri:ResponseTimestamp>2013-09-09T16:55:24-08:00</siri:ResponseTimestamp>
            <dataObjects>
                <ServiceFrame id="SF" version="any">
                    <lines>
                        <Line version="any" id="BA:BAY PT/SFIA">
                            <Name>Pittsburg/Bay Point to San Francisco International Airport</Name>
                            <TransportMode>rail</TransportMode>
                            <PublicCode/></PublicCode>
                            <SiriLineRef>722</SiriLineRef>
                            <OperatorRef ref="BA"/>
                            <Monitored>true</Monitored>
                        </Line>
                    </lines>
                </ServiceFrame>
            </dataObjects>
        </DataObjectDelivery>
    </siri:ServiceDelivery>
</siri:Siri>
A.1.3 Example Transit Stop Response (XML)

```xml
<?xml version="1.0" encoding="utf-8"?>
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
http://www.kizoom.com/standards/netex/schema/0.99.1/xsd/NeTEx_siri.xsd" version="1.0">
  <siri:ServiceDelivery>
    <siri:ResponseTimestamp>2019-11-22T13:58:57-08:00</siri:ResponseTimestamp>
    <DataObjectDelivery>
      <siri:ResponseTimestamp>2019-11-22T13:58:57-08:00</siri:ResponseTimestamp>
      <dataObjects>
        <ServiceFrame version="any" id="SB">
          <ScheduledStopPoints>
            <ScheduledStopPoint version="any" id="890001">
              <Extensions>
                <LocationType>0</LocationType>
                <PlatformCode />
                <ParentStation>2455444</ParentStation>
              </Extensions>
              <Name>San Francisco Ferry Building (Gate E)</Name>
              <Location>
                <Longitude>-122.391612</Longitude>
                <Latitude>37.795106</Latitude>
              </Location>
            </ScheduledStopPoint>
            <ScheduledStopPoint version="any" id="890002">
              <Extensions>
                <LocationType>0</LocationType>
                <PlatformCode />
                <ParentStation>2455444</ParentStation>
              </Extensions>
              <Name>San Francisco Ferry Building (Gate G)</Name>
              <Location>
                <Longitude>-122.39079</Longitude>
                <Latitude>37.79436</Latitude>
              </Location>
            </ScheduledStopPoint>
            <ScheduledStopPoint version="any" id="2455444">
              <Extensions>
                <LocationType>1</LocationType>
                <PlatformCode />
                <ParentStation />
              </Extensions>
              <Name>San Francisco Ferry Building</Name>
              <Location>
                <Longitude>-122.3933798907</Longitude>
                <Latitude>37.7954865278</Latitude>
              </Location>
            </ScheduledStopPoint>
          </ScheduledStopPoints>
        </ServiceFrame>
      </DataObjectDelivery>
    </siri:ResponseTimestamp>
  </siri:ServiceDelivery>
</siri:Siri>
```
A.1.4 Example Transit Stop Place Response (XML)

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
<siri:Siri xsi:schemaLocation="http://www.siri.org.uk/siri ../../../xsd/NeTEx_siri.xsd"
xmlns:siri="http://www.siri.org.uk/siri"
xmlns="http://www.netex.org.uk/netex"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:atom="http://www.w3.org/2005/Atom"
version="1.0">
   <siri:ServiceDelivery>
      <siri:ResponseTimestamp>2012-12-17T09:30:46-05:00</siri:ResponseTimestamp>
      <DataObjectDelivery>
         <siri:ResponseTimestamp>2012-12-17T09:30:47.0Z</siri:ResponseTimestamp>
         <dataObjects>
            <SiteFrame version="any" id="SF">
               <stopPlaces>
                  <StopPlace version="01" id="BA:12232">
                     <Name>BART LAKE MERRIT</Name>
                     <Description>800 Madison StreetOakland, CA 94607 (Between Madison St & Fallon St and 8th & 9th)</Description>
                     <Centroid>
                        <Location>
                           <Longitude>-122.265668</Longitude>
                           <Latitude>37.797345</Latitude>
                        </Location>
                     </Centroid>
                     <AccessibilityAssessment version="any" id="AccessibilityAssessment:BA:12232">
                        <MobilityImpairedAccess>true</MobilityImpairedAccess>
                        <limitations>
                           <AccessibilityLimitation>
                              <WheelchairAccess>true</WheelchairAccess>
                           </AccessibilityLimitation>
                        </limitations>
                     </AccessibilityAssessment>
                     <alternativeNames>
                        <AlternativeName version="any" id="AlternativeName:BA:12232">
                           <Name>Lake Merrit Station</Name>
                        </AlternativeName>
                     </alternativeNames>
                  </StopPlace>
               </stopPlaces>
            </SiteFrame>
         </dataObjects>
      </DataObjectDelivery>
   </siri:ServiceDelivery>
</siri:Siri>
```

</AlternativeName>
</alternativeNames>
<PostalAddress version="any" id="PostalAddress:BA:12232">
  <AddressLine1>800 Madison St</AddressLine1>
  <Town>Oakland</Town>
</PostalAddress>
<Url>http://www.bart.gov/stations/LAKE</Url>
<OperatorRef ref="BA"/>

<adjacentSites>
  <ParkingRef ref="4234"/>
</adjacentSites>

<placeEquipments>
  <SanitaryEquipment version="any" id="123">
    <Description>RestRoom in upper level</Description>
  </SanitaryEquipment>
  <CycleStorageEquipment version="any" id="233">
    <Description>Bike Racks</Description>
    <CycleStorageType>racks</CycleStorageType>
    <NumberOfSpaces>4</NumberOfSpaces>
  </CycleStorageEquipment>
  <CycleStorageEquipment version="any" id="242">
    <Description>Bike Lockers</Description>
    <CycleStorageType>other</CycleStorageType>
    <NumberOfSpaces>10</NumberOfSpaces>
  </CycleStorageEquipment>
  <SignEquipment version="any" id="141">
    <Description>Information Display Board</Description>
  </SignEquipment>
  <EscalatorEquipment version="any" id="335">
    <Description>Escalator 335</Description>
  </EscalatorEquipment>
  <LiftEquipment version="any" id="312">
    <Description>Escalator 312</Description>
  </LiftEquipment>
  <ShelterEquipment version="any" id="12">
    <Description>Waiting area 1</Description>
  </ShelterEquipment>
  <SeatingEquipment version="any" id="4566">
    <Description>Bench near waiting area</Description>
  </SeatingEquipment>
</placeEquipments>

<PublicCode>1564</PublicCode>
<TransportMode>rail</TransportMode>
<StopPlaceType>railStation</StopPlaceType>
<quays>
  <Quay version="any" id="543">
    <CompassOctant>W</CompassOctant>
  </Quay>
</quays>
</StopPlace>
</stopPlaces>
<parkings>
  <Parking version="any" id="4234">
    <Name>Lake Merritt BART Station Parking</Name>
    <Description>On Broadway, between 11th &amp; 14th</Description>
    <Centroid>
      <Location>
        <Longitude>-122.266382</Longitude>
        <Latitude>37.796615</Latitude>
      </Location>
    </Centroid>
  </Parking>
</parkings>
A.1.5 Example Transit Pattern Response (XML)

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
```

```xml
</Location>
</Centroid>
  <PostalAddress version="any" id="PostalAddress:4234">
    <AddressLine1>800 Madison St</AddressLine1>
    <Town>Oakland</Town>
  </PostalAddress>
  <ParkingType>trainStationParking</ParkingType>
  <TotalCapacity>296</TotalCapacity>
  <RealTimeOccupancyAvailable>false</RealTimeOccupancyAvailable>
</parkingAreas>
  <ParkingArea version="any" id="123">
    <Description>Accessible Parking</Description>
    <ParkingProperties>
      <ParkingUserType>registeredDisabled</ParkingUserType>
      <spaces>
        <ParkingCapacity version="any" id="ParkingCapacity:123">
          <NumberOfSpaces>10</NumberOfSpaces>
        </ParkingCapacity>
      </spaces>
    </ParkingProperties>
  </ParkingArea>
  <ParkingArea version="any" id="124">
    <Description>Reserved Parking</Description>
    <ParkingProperties>
      <ParkingUserType>reservationHolders</ParkingUserType>
      <spaces>
        <ParkingCapacity version="any" id="ParkingCapacity:124">
          <NumberOfSpaces>99</NumberOfSpaces>
        </ParkingCapacity>
      </spaces>
    </ParkingProperties>
  </ParkingArea>
</parkings>
</SiteFrame>
</dataObjects>
</DataObjectDelivery>
</siri:ServiceDelivery>
</siri:Siri>
```
<siri:Siri version="1.0" xmlns:gml="http://www.opengis.net/gml"
xmlns:siri="http://www.siri.org.uk/siri"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  <siri:ResponseTimestamp>2016-11-08T09:41:50-08:00</siri:ResponseTimestamp>
  <DataObjectDelivery>
    <siri:ResponseTimestamp>2016-11-08T09:41:50-08:00</siri:ResponseTimestamp>
    <dataObjects>
      <ServiceFrame id="Alcatraz Hornblower Ferry" version="any">
        <directions>
          <Direction id="IB" version="any">
            <Name>Inbound</Name>
          </Direction>
          <Direction id="OB" version="any">
            <Name>Outbound</Name>
          </Direction>
        </directions>
        <journeyPatterns>
          <ServiceJourneyPattern id="192989" version="any">
            <Extensions>
              <LineRef version="any" ref="Day Tour Ferry"/>
            </Extensions>
            <DirectionRef version="any" ref="IB"/>
            <DestinationDisplayView>
              <FrontText>Alcatraz</FrontText>
            </DestinationDisplayView>
            <pointsInSequence>
              <TimingPointInJourneyPattern id="8329124" version="any" order="1">
                <ScheduledStopPointRef version="any" ref="12175093"/>
                <Extensions>
                  <Name>Pier 33</Name>
                </Extensions>
              </TimingPointInJourneyPattern>
              <TimingPointInJourneyPattern id="8329125" version="any" order="2">
                <ScheduledStopPointRef version="any" ref="12175092"/>
                <Extensions>
                  <Name>Alcatraz</Name>
                </Extensions>
              </TimingPointInJourneyPattern>
            </pointsInSequence>
          </ServiceJourneyPattern>
          <ServiceJourneyPattern id="192990" version="any">
            <Extensions>
              <LineRef version="any" ref="Day Tour Ferry"/>
            </Extensions>
            <DirectionRef version="any" ref="OB"/>
            <DestinationDisplayView>
              <FrontText>Pier 33</FrontText>
            </DestinationDisplayView>
            <pointsInSequence>
              <TimingPointInJourneyPattern id="8329126" version="any" order="1">
                <ScheduledStopPointRef version="any" ref="12175092"/>
                <Extensions>
                  <Name>Alcatraz</Name>
                </Extensions>
              </TimingPointInJourneyPattern>
              <TimingPointInJourneyPattern id="8329127" version="any" order="2">
                <Extensions>
                  <Name>Alcatraz</Name>
                </Extensions>
              </TimingPointInJourneyPattern>
            </pointsInSequence>
          </ServiceJourneyPattern>
        </journeyPatterns>
      </ServiceFrame>
    </dataObjects>
  </DataObjectDelivery>
</siri:Siri>
<ScheduledStopPointRef version="any" ref="12175093"/>
<Extensions>
  <Name>Pier 33</Name>
</Extensions>
</TimingPointInJourneyPattern>
</pointsInSequence>
</ServiceJourneyPattern>
</journeyPatterns>
</ServiceFrame>
</dataObjects>
</DataObjectDelivery>
</siri:ServiceDelivery>
</siri:Siri>

### A.1.6 Example Timetable Response (XML)

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
  <siri:ServiceDelivery>
    <siri:ResponseTimestamp>2012-12-17T09:30:46-05:00</siri:ResponseTimestamp>
    <DataObjectDelivery>
      <siri:ResponseTimestamp>2012-12-17T09:30:47.0Z</siri:ResponseTimestamp>
      <dataObjects>
        <CompositeFrame id="CF" version="1">
          <frames>
            <ServiceFrame id="SF" version="any">
              <routes>
                <Route id="BG:TIBURON:North:Weekday" version="any">
                  <Name>Tiburon_North_Weekday</Name>
                  <LineRef ref="BG:TIBURON" version="any"/>
                  <DirectionRef ref="BG:TIBURON:North" version="any"/>
                  <pointsInSequence>
                    <PointOnRoute id="BG:TIBURON:North:Weekday:1" version="any">
                      <PointRef ref="BG:4432" version="any" xsi:type="ScheduledStopPointRefStructure"/>
                    </PointOnRoute>
                    <PointOnRoute id="BG:TIBURON:North:Weekday:2" version="any">
                      <PointRef ref="BG:4433" version="any" xsi:type="ScheduledStopPointRefStructure"/>
                    </PointOnRoute>
                    <PointOnRoute id="BG:TIBURON:North:Weekday:3" version="any">
                      <PointRef ref="BG:4437" version="any" xsi:type="ScheduledStopPointRefStructure"/>
                    </PointOnRoute>
                  </pointsInSequence>
                </Route>
              </routes>
              <Route id="BG:TIBURON:North:Weekend" version="any">
                <Name>Tiburon_North_Weekend</Name>
                <LineRef ref="BG:TIBURON" version="any"/>
                <DirectionRef ref="BG:TIBURON:North" version="any"/>
                <pointsInSequence>
                  <PointOnRoute id="BG:TIBURON:North:Weekend:1" version="any"/>
                  <PointOnRoute id="BG:TIBURON:North:Weekend:2" version="any"/>
                  <PointOnRoute id="BG:TIBURON:North:Weekend:3" version="any"/>
                </pointsInSequence>
              </Route>
            </ServiceFrame>
          </frames>
        </CompositeFrame>
      </dataObjects>
    </DataObjectDelivery>
  </siri:ServiceDelivery>
</siri:Siri>
```
<PointRef ref="BG:4433" version="any" xsi:type="ScheduledStopPointRefStructure" />
</PointOnRoute>
<PointOnRoute id="BG:TIBURON:North:Weekend:2" version="any">
<PointRef ref="BG:4437" version="any" xsi:type="ScheduledStopPointRefStructure" />
</PointOnRoute>
</routes>
</ServiceCalendarFrame>
<ServiceFrame id="SC" version="any">
<dayTypes>
<DayType id="BG:Weekday" version="any">
<Name>Weekday</Name>
<properties>
<PropertyOfDay>
<DaysOfWeek>Monday Tuesday Wednesday Thursday Friday</DaysOfWeek>
</PropertyOfDay>
</properties>
</DayType>
<DayType id="BG:Weekend" version="any">
<Name>Weekend</Name>
<properties>
<PropertyOfDay>
<DaysOfWeek>Saturday Sunday</DaysOfWeek>
</PropertyOfDay>
</properties>
</DayType>
</dayTypes>
<dayTypeAssignments>
<DayTypeAssignment>
<DayTypeRef ref="BG:Weekday" version="any" />
</DayTypeAssignment>
<dayTypeAssignments>
</ServiceCalendarFrame>
<TimetableFrame id="BG:TIBURON:North:Weekday" version="any">
<Name>Tiburon_North_Weekday</Name>
<frameValidityConditions>
<AvailabilityCondition id="AC:BG:TIBURON:North:Weekday" version="any">
<FromDate>2013-02-06T00:00Z</FromDate>
<ToDate>2013-06-06T00:00Z</ToDate>
<dayTypes>
<DayTypeRef ref="BG:Weekday" version="any" />
</dayTypes>
</AvailabilityCondition>
</frameValidityConditions>
<vehicleJourneys>
<ServiceJourney id="BG:11455" version="any">
<SiriVehicleJourneyRef>11455</SiriVehicleJourneyRef>
<JourneyPatternView>
<ServiceJourneyPatternRef ref="BG:112333" version="any" />
<RouteRef ref="BG:TIBURON:North:Weekday" version="any" />
<DirectionRef ref="BG:TIBURON:North" version="any" />
</JourneyPatternView>
<calls>
<Call order="1">
<ScheduledStopPointRef ref="BG:4432" />
<Arrival>

<Time>06:05:00</Time>
<Time>06:05:00</Time>
<Time>06:05:00</Time>
<Time>06:05:00</Time>
<Time>06:30:00</Time>
<Time>06:30:00</Time>
<Time>06:30:00</Time>
<Time>06:30:00</Time>
<Time>10:10:00</Time>
<Time>10:10:00</Time>
<Time>10:10:00</Time>
<Time>10:10:00</Time>
<Time>10:45:00</Time>
<Time>10:45:00</Time>
<Time>10:45:00</Time>
<Time>10:45:00</Time>
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4433" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
<ScheduledStopPointRef ref="BG:4437" />
A.1.7 Example Transit Holiday Response (XML)

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
<siri:Siri xmlns:siri="http://www.siri.org.uk/siri"
http://www.kizoom.com/standards/netex/schema/0.99.1/xsd/NeTEx_siri.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns=http://www.netex.org.uk/netex" xmlns:gml="http://www.opengis.net/gml" version="1.0">
<siri:ServiceDelivery>
<siri:ResponseTimestamp>2017-09-21T14:19:54-07:00</siri:ResponseTimestamp>
<DataObjectDelivery>
<siri:ResponseTimestamp>2017-09-21T14:19:54:07:00</siri:ResponseTimestamp>
<dataObjects>
<ServiceCalendarFrame id="SB" version="any">
<ServiceCalendar id="SB" version="any">
<FromDate>2017-05-01</FromDate>
<ToDate>2017-10-29</ToDate>
</ServiceCalendar>
</ServiceCalendarFrame>
<contentValidityConditions>
<AvailabilityConditions version="any" id="SB:2017-07-04">
<FromDate>2017-07-04T00:00:00-07:00</FromDate>
<ToDate>2017-07-04T23:59:00-07:00</ToDate>
</AvailabilityConditions>
<AvailabilityConditions version="any" id="SB:2017-09-04">
<FromDate>2017-09-04T00:00:00-07:00</FromDate>
<ToDate>2017-09-04T23:59:00-07:00</ToDate>
</AvailabilityConditions>
</contentValidityConditions>
</ServiceCalendarFrame>
</dataObjects>
</siri:ServiceDelivery>
</siri:Siri>
```

A.1.8 Example Transit Announcement Response (XML)

```xml
<?xml version="1.0" encoding="utf-8"?>
<ServiceDelivery>
<ResponseTimestamp>2013-02-14T16:05:51Z</ResponseTimestamp>
<SituationExchangeDelivery version="1.3">
<ResponseTimestamp>2013-02-14T16:05:51Z</ResponseTimestamp>
<Situations>
<PtSituationElement>
<creationTime>2013-02-14T16:00:01Z</CreationTime>
<SituationNumber>734</SituationNumber>
</PtSituationElement>
</Situations>
</SituationExchangeDelivery>
</ServiceDelivery>
```
<ValidityPeriod>
  <StartTime>2013-02-14T16:00:00Z</StartTime>
  <EndTime>2013-02-14T18:00:00Z</EndTime>
</ValidityPeriod>

<Priority>1</Priority>

<ScopeType>route</ScopeType>

<Summary>Major BART Delay</Summary>

<Description>On Thursday, February 14, at 4:00pm, BART reports a major delay on the Daly City Line in the East Bay direction due to an equipment problem on a train.</Description>

<InfoLinks>
  <InfoLink>
    <Uri>http://www.bart.gov/</Uri>
  </InfoLink>
</InfoLinks>

<Consequences>
  <Severity>severe</Severity>
  <Affects>
    <Operators>
      <AffectedOperator>
        <OperatorRef>BA</OperatorRef>
        <OperatorName>BART</OperatorName>
      </AffectedOperator>
    </Operators>
    <Networks>
      <AffectedNetwork>
        <AffectedLine>
          <LineRef>05099</LineRef>
        </AffectedLine>
      </AffectedNetwork>
    </Networks>
    <StopPoints>
      <AffectedStopPoints>
        <StopPointRef>198761</StopPointRef>
        <StopPointRef>198762</StopPointRef>
        <StopPointRef>198763</StopPointRef>
        <StopPointRef>198764</StopPointRef>
      </AffectedStopPoints>
    </StopPoints>
  </Affects>
</Consequences>
</PtSituationElement>
</Situations>
</SituationExchangeDelivery>
</ServiceDelivery>
</Siri>
A.1.10 Example Transit Real Time Predictions at a Stop Response (XML) in SIRI format

```xml
<?xml version="1.0" encoding="UTF-8"?>

<StopTimetableDelivery>
  <ResponseTimestamp>2013-09-10T13:08:23-08:00</ResponseTimestamp>
  <SubscriptionRef>511SFBay</SubscriptionRef>
  <TimetabledStopVisit version="1.4">
    <RecordedAtTime>2013-09-02T22:16:20-08:00</RecordedAtTime>
    <MonitoringRef>12018522</MonitoringRef>
    <TargetedVehicleJourney>
      <LineRef>917</LineRef>
      <DirectionRef>S</DirectionRef>
      <DataFrameRef>2013-08-22</DataFrameRef>
      <DatedVehicleJourneyRef>4718334</DatedVehicleJourneyRef>
      <PublishedLineName>DALY/FREMONT</PublishedLineName>
      <OperatorRef>BA</OperatorRef>
      <OriginRef>12018513</OriginRef>
      <OriginName>BART DALY CITY</OriginName>
      <DestinationRef>12018519</DestinationRef>
      <DestinationName>BART FREMONT</DestinationName>
      <VehicleJourneyName>FREMONT</VehicleJourneyName>
      <VisitNumber>1</VisitNumber>
      <AimedArrivalTime>2013-08-22T12:01:00</AimedArrivalTime>
      <AimedDepartureTime>2013-08-22T12:01:00</AimedDepartureTime>
    </TargetedCall>
    </TargetedVehicleJourney>
  </TimetabledStopVisit>
  <TimetabledStopVisit version="1.4">
    <RecordedAtTime>2013-09-02T22:16:20-08:00</RecordedAtTime>
    <MonitoringRef>12018522</MonitoringRef>
    <TargetedVehicleJourney>
      <LineRef>917</LineRef>
      <DirectionRef>S</DirectionRef>
      <DataFrameRef>2013-08-22</DataFrameRef>
      <DatedVehicleJourneyRef>4718335</DatedVehicleJourneyRef>
      <PublishedLineName>DALY/FREMONT</PublishedLineName>
      <OperatorRef>BA</OperatorRef>
      <OriginRef>12018513</OriginRef>
      <OriginName>BART DALY CITY</OriginName>
      <DestinationRef>12018519</DestinationRef>
      <DestinationName>BART FREMONT</DestinationName>
      <VehicleJourneyName>FREMONT</VehicleJourneyName>
      <VisitNumber>1</VisitNumber>
      <AimedArrivalTime>2013-08-22T12:16:00</AimedArrivalTime>
      <AimedDepartureTime>2013-08-22T12:16:00</AimedDepartureTime>
    </TargetedCall>
    </TargetedVehicleJourney>
  </TimetabledStopVisit>
</StopTimetableDelivery>
```

A.1.10 Example Transit Real Time Predictions at a Stop Response (XML) in SIRI format
  <ServiceDelivery>
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>BA</ProducerRef>
    <Status>true</Status>
    <StopMonitoringDelivery version="1.4">
      <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
      <Status>true</Status>
      <MonitoredStopVisit>
        <RecordedAtTime>2004-12-17T09:25:46-05:00</RecordedAtTime>
        <MonitoringRef>EMBR</MonitoringRef>
        <MonitoredVehicleJourney>
          <LineRef>Warm Springs/South Fremont - Daly City</LineRef>
          <DirectionRef>E</DirectionRef>
          <FramedVehicleJourneyRef>
            <DataFrameRef>2004-12-17</DataFrameRef>
            <DatedVehicleJourneyRef>1031357WKDY</DatedVehicleJourneyRef>
          </FramedVehicleJourneyRef>
          <PublishedLineName>Warm Springs/South Fremont - Daly City</PublishedLineName>
          <OperatorRef>BA</OperatorRef>
          <OriginRef>DALY</OriginRef>
          <OriginName>Daly City BART Station</OriginName>
          <DestinationRef>WARM</DestinationRef>
          <DestinationName>Warm Springs/South Fremont</DestinationName>
          <Monitored>true</Monitored>
          <InCongestion>false</InCongestion>
          <VehicleLocation>
            <Longitude>180</Longitude>
            <Latitude>90</Latitude>
          </VehicleLocation>
          <Bearing>23</Bearing>
          <Occupancy>false</Occupancy>
          <VehicleRef>1011730</VehicleRef>
          <ProgressStatus>Service running on time</ProgressStatus>
          <PreviousCalls>
            <PreviousCall>
              <StopPointRef>BART_10</StopPointRef>
              <VisitNumber>2</VisitNumber>
              <StopPointName>BART_DALY CITY</StopPointName>
              <VehicleAtStop>false</VehicleAtStop>
              <AimedDepartureTime>2004-12-17T09:32:43-05:00</AimedDepartureTime>
              <ActualDepartureTime>2004-12-17T09:32:43-05:00</ActualDepartureTime>
            </PreviousCall>
          </PreviousCalls>
          <MonitoredCall>
            <StopPointRef>EMBR</StopPointRef>
            <VisitNumber>1</VisitNumber>
            <StopPointName>Embarcadero BART Station</StopPointName>
            <VehicleAtStop>false</VehicleAtStop>
            <VehicleLocationAtStop>
              <Longitude>180</Longitude>
              <Latitude>90</Latitude>
            </VehicleLocationAtStop>
            <AimedArrivalTime>2004-12-17T09:40:46-05:00</AimedArrivalTime>
            <ExpectedArrivalTime>2004-12-17T09:40:46-05:00</ExpectedArrivalTime>
          </MonitoredCall>
        </MonitoredVehicleJourney>
      </MonitoredStopVisit>
    </StopMonitoringDelivery>
  </ServiceDelivery>
</Siri>
<AimedDepartureTime>2004-12-17T09:42:47-05:00</AimedDepartureTime>
<ExpectedDepartureTime>2004-12-17T09:40:47-05:00</ExpectedDepartureTime>

</MonitoredCall>
</OnwardCalls>

<OnwardCall>
  <StopPointRef>BART_12</StopPointRef>
  <VisitNumber>4</VisitNumber>
  <StopPointName>BAR_12th St Oakland</StopPointName>
  <VehicleAtStop>false</VehicleAtStop>
  <AimedArrivalTime>2004-12-17T09:30:56-05:00</AimedArrivalTime>
  <ExpectedArrivalTime>2004-12-17T09:30:56-05:00</ExpectedArrivalTime>
</OnwardCall>

</MonitoredVehicleJourney>

<MonitoredStopVisit>
  <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
  <ItemRef>SED9843214675429</ItemRef>
  <Reason>Arrived</Reason>
</MonitoredStopVisit>

<MonitoredStopVisitCancellation>
  <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
  <MonitoringRef>BART_11</MonitoringRef>
  <VisitNumber>2</VisitNumber>
  <LineRef>Line123</LineRef>
  <DirectionRef>OB</DirectionRef>
  <VehicleJourneyRef>
    <DataFrameRef>2004-12-17</DataFrameRef>
    <DatedVehicleJourneyRef>0987656</DatedVehicleJourneyRef>
  </VehicleJourneyRef>
  <Reason>Arrived</Reason>
</MonitoredStopVisitCancellation>

<StopLineNotice>
  <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
  <ItemIdentifier>SED9843214675429</ItemIdentifier>
  <MonitoringRef>BART_11</MonitoringRef>
  <LineRef>123</LineRef>
  <DirectionRef>OB</DirectionRef>
  <LineNote>Mechanical Problems on Track</LineNote>
</StopLineNotice>

<StopLineNoticeCancellation>
  <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
  <ItemRef>SED9843214675429</ItemRef>
  <MonitoringRef>BART_11</MonitoringRef>
  <LineRef>123</LineRef>
  <DirectionRef>OB</DirectionRef>
</StopLineNoticeCancellation>

</StopMonitoringDelivery>
</ServiceDelivery>
</Siri>
<?xml version="1.0" encoding="UTF-8"?>
  <ServiceDelivery>
    <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
    <ProducerRef>BA</ProducerRef>
    <Status>true</Status>
    <VehicleMonitoringDelivery version="1.4">
      <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
      <VehicleActivity>
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
        <ValidUntilTime>2004-12-17T09:30:47-05:00</ValidUntilTime>
        <MonitoredVehicleJourney>
          <LineRef>17</LineRef>
          <DirectionRef>OB</DirectionRef>
          <FramedVehicleJourneyRef>
            <DataFrameRef>2004-12-17</DataFrameRef>
            <DatedVehicleJourneyRef>987675</DatedVehicleJourneyRef>
          </FramedVehicleJourneyRef>
          <PublishedLineName>123</PublishedLineName>
          <OperatorRef>BA</OperatorRef>
          <OriginName>SFO</OriginName>
          <Via>
            <PlaceName>16th st</PlaceName>
          </Via>
          <Via>
            <PlaceName>West Oakland</PlaceName>
          </Via>
          <DestinationRef>Fremont</DestinationRef>
          <DestinationName>Fremont</DestinationName>
          <Monitored>true</Monitored>
          <InCongestion>false</InCongestion>
          <VehicleLocation>
            <Longitude>180</Longitude>
            <Latitude>90</Latitude>
          </VehicleLocation>
          <Bearing>123</Bearing>
          <Occupancy>full</Occupancy>
          <ProgressRate>slowProgress</ProgressRate>
          <Delay>PT2M</Delay>
          <ProgressStatus>On time</ProgressStatus>
          <VehicleRef>VEH987654</VehicleRef>
        </MonitoredVehicleJourney>
        <PreviousCalls>
          <PreviousCall>
            <StopPointRef>SFO</StopPointRef>
            <VisitNumber>2</VisitNumber>
            <StopPointName>String</StopPointName>
            <VehicleAtStop>false</VehicleAtStop>
            <AimedDepartureTime>2004-12-17T09:32:43-05:00</AimedDepartureTime>
            <ActualDepartureTime>2004-12-17T09:32:43-05:00</ActualDepartureTime>
          </PreviousCall>
        </PreviousCalls>
        <OnwardCalls>
          <OnwardCall>
            <StopPointRef>80</StopPointRef>
            <VisitNumber>4</VisitNumber>
          </OnwardCall>
        </OnwardCalls>
      </VehicleActivity>
    </VehicleMonitoringDelivery>
  </ServiceDelivery>
</Siri>
A.1.12 Example Transit Schedule Update Response (XML) in SIRI PT format

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <MonitoredVehicleJourney>
    <VehicleActivity>
      <OnwardCalls>
        <OnwardCall>
          <StopPointRef>HLTST012</StopPointRef>
          <StopPointName>Church</StopPointName>
        </OnwardCall>
      </OnwardCalls>
    </VehicleActivity>
    <VehicleActivityCancellation>
      <RecordedAtTime>2004-12-17T09:30:47:05:00</RecordedAtTime>
      <VehicleMonitoringRef>9876542</VehicleMonitoringRef>
      <VehicleJourneyRef>
        <DataFrameRef>2001-12-17</DataFrameRef>
        <DatedVehicleJourneyRef>09867</DatedVehicleJourneyRef>
      </VehicleJourneyRef>
      <LineRef>Line123</LineRef>
      <DirectionRef>OB</DirectionRef>
      <Reason>Done for the day</Reason>
      <VehicleActivityCancellation>
        <VehicleMonitoringDelivery>
          <ServiceDelivery/>
          </VehicleMonitoringDelivery>
        </VehicleActivityCancellation>
      </VehicleActivity>
    </Monitor>
  </MonitoredVehicleJourney>
</Siri>
```
A.1.13 Example Transit Addition and Cancellation of Trip Response (XML) in SIRI ET format

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <ServiceDelivery>
    <ResponseTimestamp>2013-02-18T09:30:47-08:00</ResponseTimestamp>
    <Status>true</Status>
    <ProductionTimetableDelivery version="1.4">
      <ResponseTimestamp>2004-12-17T09:30-05:00</ResponseTimestamp>
      <ValidUntil>2001-12-17T10:47-05:00</ValidUntil>
    </ProductionTimetableDelivery>
    <DatedTimetableVersionFrame>
      <RecordedAtTime>2001-12-17T09:30-05:00</RecordedAtTime>
      <LineRef>123</LineRef>
      <DirectionRef>Out</DirectionRef>
      <PublishedLineName>String</PublishedLineName>
      <DatedVehicleJourney>
        <DatedCalls>
          <DatedCall>
            <StopPointRef>BART_11</StopPointRef>
            <CallNote>optional message here</CallNote>
            <AimedArrivalTime>2013-02-19T09:55-08:00</AimedArrivalTime>
            <AimedDepartureTime>2013-02-19T09:56-08:00</AimedDepartureTime>
          </DatedCall>
          <DatedCall>
            <StopPointRef>BART_99</StopPointRef>
            <CallNote>optional message here</CallNote>
            <AimedArrivalTime>2013-02-19T10:15-08:00</AimedArrivalTime>
            <AimedDepartureTime>2013-02-19T10:16-08:00</AimedDepartureTime>
          </DatedCall>
        </DatedCalls>
        <DatedVehicleJourney>...
      </DatedVehicleJourney>
    </DatedTimetableVersionFrame>
    <ProductionTimetableDelivery>
    </ProductionTimetableDelivery>
  </ServiceDelivery>
</Siri>
```
A.1.14 Example Transit General Messaging Service Response (XML) in SIRI GM format

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <ServiceDelivery>
    <ResponseTimestamp>2013-02-17T09:30:46-08:00</ResponseTimestamp>
    <Status>true</Status>
    <GeneralMessageDelivery version="1.4">
      <ResponseTimestamp>2001-12-17T09:30:47.0Z</ResponseTimestamp>
      <GeneralMessage formatRef="string">
        <RecordedAtTime>2013-02-17T09:30:46-08:00</RecordedAtTime>
        <InfoMessageIdentifier>12345</InfoMessageIdentifier>
        <InfoMessageVersion>2</InfoMessageVersion>
        <InfoChannelRef>WARNINGS</InfoChannelRef>
        <ValidUntilTime>2013-02-18T09:30:46-08:00</ValidUntilTime>
        <Content>some message here</Content>
      </GeneralMessage>
      <GeneralMessage formatRef="string">
        <RecordedAtTime>2013-02-17T09:30:46-08:00</RecordedAtTime>
        <InfoMessageIdentifier>23456</InfoMessageIdentifier>
        <InfoMessageVersion>1</InfoMessageVersion>
        <InfoChannelRef>WARNINGS</InfoChannelRef>
        <ValidUntilTime>2013-02-18T09:30:46-08:00</ValidUntilTime>
        <Content>some message here</Content>
      </GeneralMessage>
    </GeneralMessageDelivery>
  </ServiceDelivery>
</Siri>
```
A.1.15 Example Transit GTFS Operator List in XML format

```xml
<GTFSAgencies>
    <GTFSAgency Id="3D" Name="Tri Delta Transit" LastGenerated="9/8/2017 5:22:04 PM"/>
    <GTFSAgency Id="AC" Name="AC Transit" LastGenerated="8/27/2017 6:06:13 PM"/>
    <GTFSAgency Id="BA" Name="BART" LastGenerated="6/6/2017 1:26:30 PM"/>
    <GTFSAgency Id="CC" Name="County Connection" LastGenerated="8/16/2016 2:04:53 PM"/>
    <GTFSAgency Id="CT" Name="Caltrain" LastGenerated="6/20/2017 11:04:29 AM"/>
    <GTFSAgency Id="EM" Name="Emery Go-Round" LastGenerated="8/8/2017 4:50:45 PM"/>
    <GTFSAgency Id="GF" Name="Golden Gate Ferry" LastGenerated="7/26/2017 11:27:08 AM"/>
    <GTFSAgency Id="RG" Name="Regional GTFS" LastGenerated="6/15/2020 5:11:56 AM"/>
    <GTFSAgency Id="MA" Name="Marin Transit" LastGenerated="8/16/2016 1:54:42 PM"/>
</GTFSAgencies>
```

A.1.16 Example Transit ServiceAlerts Response (XML)

```xml
<FeedMessage xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
              xmlns="http://schemas.datacontract.org/2004/07/TransitRealtime">
    <Entities>
        <FeedEntity>
            <Alert>
                <ActivePeriods>
                    <TimeRange>
                        <End>1609488000</End>
                        <Start>1532736000</Start>
                    </TimeRange>
                    <ActivePeriods/>
                </ActivePeriods>
                <DescriptionText>
                    <Translations>
                        <TranslatedString.Translation>
                            <Language>en</Language>
                            <Text>Call or visit 511.org for more real-time departures and alert information.</Text>
                        </TranslatedString.Translation>
                    </Translations>
                </DescriptionText>
                <HeaderText>
                    <Translations>
                        <TranslatedString.Translation>
                            <Language>en</Language>
                            <Text>Call 511 or visit 511.org for more alert information. Issues with this sign? feedback@511.org</Text>
                        </TranslatedString.Translation>
                    </Translations>
                </HeaderText>
            </Alert>
            <InformedEntities>
                <EntitySelector>
                    <AgencyId>5S</AgencyId>
                    <RouteId></RouteId>
                    <RouteType>0</RouteType>
                    <StopId></StopId>
                    <Trip i:nil="true"/>
                </EntitySelector>
            </InformedEntities>
        </FeedEntity>
    </Entities>
</FeedMessage>
```
<TtsHeaderText i:nil="true" />
<Url>
  <Translations>
    <TranslatedString.Translation>
      <Language>en</Language>
      <Text>For Issues reach out to feedback@511.org</Text>
    </TranslatedString.Translation>
  </Translations>
</Url>

<cause>UnknownCause</cause>
<effect>UnknownEffect</effect>
<severity_level>UnknownSeverity</severity_level>
</Alert>

<Id>3469</Id>
<IsDeleted>false</IsDeleted>
<TripUpdate i:nil="true" />
<Vehicle i:nil="true" />
</FeedEntity>

<ActivePeriods>
  <TimeRange>
    <End>1704049140</End>
    <Start>1583089140</Start>
  </TimeRange>
</ActivePeriods>

<DescriptionText>
  <Translations>
    <TranslatedString.Translation>
      <Language>en</Language>
      <Text>For Issues reach out to feedback@511.org</Text>
    </TranslatedString.Translation>
  </Translations>
</DescriptionText>

<HeaderText>
  <Translations>
    <TranslatedString.Translation>
      <Language>en</Language>
      <Text>Issues? feedback@511.org</Text>
    </TranslatedString.Translation>
  </Translations>
</HeaderText>

<EntitySelector>
  <AgencyId>5S</AgencyId>
  <RouteId></RouteId>
  <RouteType>0</RouteType>
  <StopId></StopId>
  <Trip i:nil="true" />  
</EntitySelector>

<TtsDescriptionText i:nil="true" />
<TtsHeaderText i:nil="true" />
<Url>
  <Translations>
    <TranslatedString.Translation>
      <Language>en</Language>
      <Text>For Issues reach out to feedback@511.org</Text>
    </TranslatedString.Translation>
  </Translations>
</Url>
<Alert>
  <Id>21538899</Id>
  <IsDeleted>false</IsDeleted>
  <TripUpdate i:nil="true" />
  <Vehicle i:nil="true" />
</Alert>
</FeedEntity>
</Entities>
</Header>
</FeedMessage>
A.1.17 Example Shapes Response (XML)

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
  <siri:ServiceDelivery>
    <siri:ResponseTimestamp>2019-11-22T10:45:00-08:00</siri:ResponseTimestamp>
    <DataObjectDelivery>
      <siri:ResponseTimestamp>2019-11-22T10:45:00-08:00</siri:ResponseTimestamp>
      <dataObjects>
        <TimetableFrame version="any" id="TF:BA">
          <vehicleJourneys>
            <ServiceJourney version="any" id="3010811SUN">
              <LinkSequenceProjection version="any" id="245016">
                <gml:LineString srsName="WGS84" gml:id="245016">
                  <gml:pos>37.60088289 -122.3951075</gml:pos>
                  <gml:pos>37.60939749 -122.3964303</gml:pos>
                  <gml:pos>37.61078442 -122.3978387</gml:pos>
                  <gml:pos>37.61178447 -122.3988195</gml:pos>
                  <gml:pos>37.61224882 -122.3992295</gml:pos>
                  <gml:pos>37.61264766 -122.3994088</gml:pos>
                  <gml:pos>37.6128218 -122.3994895</gml:pos>
                  <gml:pos>37.61297159 -122.399568</gml:pos>
                  <gml:pos>37.61317171 -122.3996879</gml:pos>
                  <gml:pos>37.61334305 -122.3997649</gml:pos>
                  <gml:pos>37.61351504 -122.3998161</gml:pos>
                  <gml:pos>37.61380724 -122.3998666</gml:pos>
                  <gml:pos>37.61393581 -122.3998685</gml:pos>
                  <gml:pos>37.61423276 -122.3998292</gml:pos>
                  <gml:pos>37.61441157 -122.3997582</gml:pos>
                  <gml:pos>37.6146182 -122.3996593</gml:pos>
                  <gml:pos>37.61478632 -122.3995339</gml:pos>
                  <gml:pos>37.61494871 -122.3993958</gml:pos>
                  <gml:pos>37.61500888 -122.3993223</gml:pos>
                  <gml:pos>37.61513029 -122.399157</gml:pos>
                  <gml:pos>37.61526027 -122.3989516</gml:pos>
                  <gml:pos>37.61533452 -122.3988807</gml:pos>
                  <gml:pos>37.61541787 -122.3985962</gml:pos>
                  <gml:pos>37.61550791 -122.3982678</gml:pos>
                  <gml:pos>37.61551978 -122.3981671</gml:pos>
                  <gml:pos>37.61556776 -122.3976527</gml:pos>
                  <gml:pos>37.61557464 -122.3974131</gml:pos>
                  <gml:pos>37.61559137 -122.3969052</gml:pos>
                  <gml:pos>37.61558998 -122.3961968</gml:pos>
                  <gml:pos>37.61560202 -122.3948619</gml:pos>
                  <gml:pos>37.6156106 -122.3947422</gml:pos>
                  <gml:pos>37.61564872 -122.3942105</gml:pos>
                  <gml:pos>37.61602033 -122.3926008</gml:pos>
                </gml:LineString>
              </LinkSequenceProjection>
            </ServiceJourney>
          </vehicleJourneys>
        </TimetableFrame>
      </dataObjects>
    </DataObjectDelivery>
  </siri:ServiceDelivery>
</siri:Siri>
```
4 Appendix B: API Response Messages- JSON

4.1 Transit JSON

B.1.1 Example Transit Operator Response (JSON)

```json
{
    "content": {
        "Id": "SF",
        "Name": "San Francisco Municipal Railway",
        "ShortName": "Muni",
        "SiriOperatorRef": "SF",
        "TimeZone": "America/Vancouver",
        "DefaultLanguage": "en",
        "ContactTelephoneNumber": "1-415-701-2311",
        "WebSite": "http://www.sfmta.com/",
        "PrimaryMode": "bus",
        "PrivateKey": "SF",
        "Monitor": "true",
        "OtherModes": "tram, funicular",
        "Coverage": {
            "type": "Polygon",
            "coordinates": [
                [47.329999999999998, -71.170000000000002],
                [47.359999999999999, -71.150000000000006],
                [47.350000000000001, -71.099999999999994],
                [47.399999999999999, -71.200000000000003],
                [47.329999999999998, -71.170000000000002]
            ]
        }
    }
}
```

B.1.2 Example Transit Line Response (JSON)
B.1.3 Example Transit Stop Response (JSON)

```json
{
  "Content": [
    {
      "Id": "BA:BAY PT/SFIA",
      "Name": "Pittsburg/Bay Point to San Francisco International Airport",
      "TransportMode": "rail",
      "SirILineRef": "722",
      "Monitored": "true",
      "OperatorRef": "BA"
    }
  ]
}
```
```
"LocationType": "0",
"PlatformCode": null,
"ParentStation": "2455444"
},
"Name": "San Francisco Ferry Building (Gate G)",
"Location": {
 "Longitude": "-122.39079",
 "Latitude": "37.79436"
 },
"Url": null,
"StopType": "ferryStop"
"
],
"stopAreas": {
 "StopArea": {
 "id": "StopArea:2455444",
 "Name": "San Francisco Ferry Building",
 "members": {
 "ScheduledStopPointRef": [
 {"ref": "890001" },
 {"ref": "890002" }
 ]
 },

 "ParentStopAreaRef": { "ref": "2455444" }

 }

 },
"StopPlace": {
  "id": "BA:12232",
  "Name": "BART LAKE MERRIT",
  "Description": "800 Madison StreetOakland, CA 94607 (Between Madison St & Fallon St and 8th & 9th)",
  "Centroid": {
   "Location": {
    "Longitude": "-122.265668",
    "Latitude": "37.797345"
    }
  },
  "AccessibilityAssessment": {
   "MobilityImpairedAccess": "true",
   "limitations": {
    "AccessibilityLimitation": { "WheelchairAccess": "true" }
    }
  },
  "alternativeNames": {
   "AlternativeName": { "Name": "Lake Merrit Station" }
  },
  "PostalAddress": {
   "AddressLine1": "800 Madison St",
   "Town": "Oakland"
  },
  "Url": "http://www.bart.gov/stations/LAKE",
  "OperatorRef": "BA",
  "StopType": "ferryStop"
}
```

B.1.4 Example Transit StopPlace Response (JSON)
"adjacentSites": { "ParkingRef": "4234" },
"placeEquipments": {
  "SanitaryEquipment": { "Description": "RestRoom in upper level" },
  "CycleStorageEquipment": [
    { "Description": "Bike Racks",
      "CycleStorageType": "racks",
      "NumberOfSpaces": "4"
    },
    { "Description": "Bike Lockers",
      "CycleStorageType": "other",
      "NumberOfSpaces": "10"
    }
  ],
  "SignEquipment": { "Description": "Information Display Board" },
  "EscalatorEquipment": { "Description": "Escalator 335" },
  "LiftEquipment": { "Description": "Escalator 312" },
  "ShelterEquipment": { "Description": "Waiting area 1" },
  "SeatingEquipment": { "Description": "Bench near waiting area" }
},
"PublicCode": "1564",
"TransportMode": "rail",
"StopPlaceType": "railStation",
"quays": {
  "Quay": { "CompassOctant": "W" }
},
"parkings": {
  "Parking": {
    "Id": "4234",
    "Name": "Lake Merritt BART Station Parking",
    "Description": "On Broadway, between 11th & 14th",
    "Centroid": {
      "Location": {
        "Longitude": ",-122.266382",
        "Latitude": "37.796615"
      }
    },
    "PostalAddress": {
      "AddressLine1": "800 Madison St",
      "Town": "Oakland"
    },
    "ParkingType": "trainStationParking",
    "TotalCapacity": "296",
    "RealTimeOccupancyAvailable": "false",
    "parkingAreas": {
      "ParkingArea": [
        { "Id": "123",
          "Description": "Accessible Parking",
          "ParkingProperties": {
            "ParkingUserType": "registeredDisabled",
            "spaces": {
              "ParkingCapacity": { "NumberOfSpaces": "10" }
            }
          }
        },
        { "Id": "124",
          "Description": "General Parking",
          "ParkingProperties": {
            "ParkingUserType": "registeredNonDisabled",
            "spaces": {
              "ParkingCapacity": { "NumberOfSpaces": "286" }
            }
          }
        }
      ]
    }
  }
}
"Description": "Reserved Parking",
"ParkingProperties": {
    "ParkingUserType": "reservationHolders",
    "spaces": {
        "ParkingCapacity": { "NumberOfSpaces": "99" }
    },
    "charges": {
        "tariffBands": [
            "ParkingTariffChargeBand": [
                { "Description": "Single Day Reserved Parking",
                 "MaximumStay": "P1D",
                 "Amount": "4.50"
                },
                { "Description": "Monthly Reserved Parking",
                 "MaximumStay": "P1M",
                 "Amount": "100"
                ]
            ]
        ]
    }
}

B.1.5 Example Transit Pattern Response (JSON)

```json
{
    "directions": [
        { "DirectionId": "IB", "Name": "Inbound" },
        { "DirectionId": "OB", "Name": "Outbound" }
    ],
    "journeyPatterns": [
        { "serviceJourneyPatternRef": "192989",
          "LineRef": "Day Tour Ferry",
          "Name": "Alcatraz",
          "DirectionRef": "IB",
          "DestinationDisplayView": { "FontText": "Alcatraz" },
          "PointsInSequence": { "StopPointInJourneyPattern": [] }
    ]
}
```
"TimingPointInJourneyPattern": [ 
  {
    "TimingPointInJourneyPatternId": "8329124",
    "Order": "1",
    "ScheduledStopPointRef": "12175093",
    "Name": "Pier 33"
  },
  {
    "TimingPointInJourneyPatternId": "8329125",
    "Order": "2",
    "ScheduledStopPointRef": "12175092",
    "Name": "Alcatraz"
  }
],
"LinksInSequence": { "ServiceLinkInJourneyPattern": "" }
},
{
  "serviceJourneyPatternRef": "192990",
  "LineRef": "Day Tour Ferry",
  "Name": "Pier 33",
  "DirectionRef": "OB",
  "DestinationDisplayView": { "FontText": "Pier 33" },
  "PointsInSequence": {
    "StopPointInJourneyPattern": [ ],
    "TimingPointInJourneyPattern": [
      {
        "TimingPointInJourneyPatternId": "8329126",
        "Order": "1",
        "ScheduledStopPointRef": "12175092",
        "Name": "Alcatraz"
      },
      {
        "TimingPointInJourneyPatternId": "8329127",
        "Order": "2",
        "ScheduledStopPointRef": "12175093",
        "Name": "Pier 33"
      }
    ]
  },
  "LinksInSequence": { "ServiceLinkInJourneyPattern": "" }
}]

B.1.6 Example Timetable Response (JSON)
{ "Content": {  "ServiceFrame": {   "id": "SF",   "routes": [    "Route": [      { "id": "86855", "Name": "10:IB:Weekdays", "LineRef": { "ref": "10" }, "DirectionRef": { "ref": "IB" }, "pointsInSequence": [        { "id": "86855:1",        "PointRef": { "ref": "17518", "type": "ScheduledStopPointRefStructure" }      },        { "id": "86855:2",        "PointRef": { "ref": "14350", "type": "ScheduledStopPointRefStructure" }      },        { "id": "86855:3",        "PointRef": { "ref": "16700", "type": "ScheduledStopPointRefStructure" }      },        { "id": "86855:4",        "PointRef": { "ref": "16695", "type": "ScheduledStopPointRefStructure" }      },        { "id": "86855:5",        "PointRef": { "ref": "16333", "type": "ScheduledStopPointRefStructure" }      ]    },    { "id": "86858", "Name": "10:OB:Weekdays", "LineRef": { "ref": "10" }, "DirectionRef": { "ref": "OB" }, "pointsInSequence": [        { "id": "86858:1",        "PointRef": { "ref": "15147", "type": "ScheduledStopPointRefStructure" }      },        { "id": "86858:2",        "PointRef": { "ref": "15150", "type": "ScheduledStopPointRefStructure" }      ]    }   ]  }  },
"PointRef": { "ref": "15859", "type": "ScheduledStopPointRefStructure" }
},
{
  "id": "86858:3",
  "PointRef": { "ref": "15853", "type": "ScheduledStopPointRefStructure" }
},
{
  "id": "86858:4",
  "PointRef": { "ref": "16327", "type": "ScheduledStopPointRefStructure" }
},
{
  "id": "86858:5",
  "PointRef": { "ref": "13008", "type": "ScheduledStopPointRefStructure" }
}
],
"ServiceCalendarFrame": {
  "id": "SF",
  "dayTypes": {
    "DayType": [
      {
        "id": "6098",
        "Name": "Weekdays",
        "properties": {
          "PropertyOfDay": { "DaysOfWeek": "Monday Tuesday Wednesday Thursday Friday"
        }
      }
    ]
  },
  "dayTypeAssignments": {
    "DayTypeAssignment": {
      "DayTypeRef": { "ref": "6098" }
    }
  }
},
"TimetableFrame": [
  {
    "id": "Timetable:86855",
    "Name": "10:IB\Weekdays",
    "frameValidityConditions": {
      "AvailabilityCondition": {
        "id": "10:IB\Weekdays",
        "FromDate": "2013-02-18T09:30:47-08:00",
        "ToDate": "2013-02-18T09:30:47-08:00",
        "dayTypes": { "DayTypeRef": { "ref": "6098" }
      }
    }
  }
}
"vehicleJourneys": {
  "ServiceJourney": [
    {
      "id": "4769819",
      "SiriVehicleJourneyRef": "4769819",
      "JourneyPatternView": {
        "RouteRef": {
          "ref": "86855"
        },
        "DirectionRef": {
          "ref": "IB"
        }
      },
      "calls": {
        "Call": [
          {
            "order": "1",
            "ScheduledStopPointRef": "17518",
            "Arrival": { "Time": "05:03:00", "DaysOffset": "0" },
            "Departure": { "Time": "05:03:00", "DaysOffset": "0" }
          },
          {
            "order": "2",
            "ScheduledStopPointRef": "14350",
            "Arrival": { "Time": "05:17:00", "DaysOffset": "0" },
            "Departure": { "Time": "05:17:00", "DaysOffset": "0" }
          },
          {
            "order": "3",
            "ScheduledStopPointRef": "16700",
            "Arrival": { "Time": "05:20:00", "DaysOffset": "0" },
            "Departure": { "Time": "05:20:00", "DaysOffset": "0" }
          },
          {
            "order": "4",
            "ScheduledStopPointRef": "16695",
            "Arrival": { "Time": "05:22:00", "DaysOffset": "0" },
            "Departure": { "Time": "05:22:00", "DaysOffset": "0" }
          },
          {
            "order": "5",
            "ScheduledStopPointRef": "16333",
            "Arrival": { "Time": "05:32:00", "DaysOffset": "0" },
            "Departure": { "Time": "05:32:00", "DaysOffset": "0" }
          }
        ]
      }
    }
  ]
}
"id": "4769820",
"SiriVehicleJourneyRef": "4769820",
"JourneyPatternView": {
  "RouteRef": { "ref": "86855"},
  "DirectionRef": {"ref": "IB"}
},
"calls": {
  "Call": [
    {
      "order": "1",
      "ScheduledStopPointRef": {"ref": "17518"},
      "Arrival": {"Time": "05:30:00", "DaysOffset": "0"},
      "Departure": {"Time": "05:30:00", "DaysOffset": "0"}
    },
    {
      "order": "2",
      "ScheduledStopPointRef": {"ref": "14350"},
      "Arrival": {"Time": "05:44:00", "DaysOffset": "0"},
      "Departure": {"Time": "05:44:00", "DaysOffset": "0"}
    },
    {
      "order": "3",
      "ScheduledStopPointRef": {"ref": "16700"},
      "Arrival": {"Time": "05:47:00", "DaysOffset": "0"},
      "Departure": {"Time": "05:47:00", "DaysOffset": "0"}
    },
    {
      "order": "4",
      "ScheduledStopPointRef": {"ref": "16695"},
      "Arrival": {"Time": "05:49:00", "DaysOffset": "0"},
      "Departure": {"Time": "05:49:00", "DaysOffset": "0"}
    },
    {
      "order": "5",
      "ScheduledStopPointRef": {"ref": "16333"},
      "Arrival": {"Time": "05:59:00", "DaysOffset": "0"},
      "Departure": {"Time": "05:59:00", "DaysOffset": "0"}
    }
  ]
}
B.1.7 Example Transit Holiday Response (JSON)

```json
{
    "Content": {
        "ServiceCalendar": {
            "id": "SB",
            "FromDate": "2017-05-01",
            "ToDate": "2017-10-29"
        },
        "AvailabilityConditions": [
            {
                "version": "any",
                "id": "SB:2017-07-04",
                "FromDate": "2017-07-04T00:00:00-07:00",
                "ToDate": "2017-07-04T23:59:00-07:00"
            },
            {
                "version": "any",
                "id": "SB:2017-09-04",
                "FromDate": "2017-09-04T00:00:00-07:00",
                "ToDate": "2017-09-04T23:59:00-07:00"
            }
        ]
    }
}
```

B.1.8 Example Transit Announcement Response (JSON)

```json
{
    "Siri": {
        "ServiceDelivery": {
            "ResponseTimestamp": "2013-09-10T15:53:47-08:00",
            "SituationExchangeDelivery": {
                "Situations": {
                    "PtSituationElement": {
                        "CreationTime": "2013-09-05T09:39:27-08:00",
                        "SituationNumber": "169230",
                        "Source": {
                            "SourceType": "feed",
                            "Name": "MTC"
                        },
                        "ValidityPeriod": {
                            "StartTime": "2013-09-05T00:00:00-08:00",
                            "EndTime": "2013-10-06T00:00:00-08:00"
                        },
                        "UnknownReason": null,
                        "Priority": "2",
                        "ScopeType": "route",
                        "Summary": "Long-term Detour on Line 74 until May 2015",
                        "Description": "Due to a long-term construction project in Richmond, Line 74 will be detoured from August 26, 2013 through May 2015. Line 74 will not serve the stops on Marina Bay Parkway at Meeker Avenue in either direction. Board Line 74 to Harbour Way on South 23rd Street at Potrero Avenue or to Hilltop Mall/Castro Ranch Road on South 23rd Street at Cutting Boulevard. Line 74 will also not serve the stops on Marina Bay Parkway at Pierson"
                        }
                    }
                }
            }
        }
    }
```

Avenue. Board Line 74 to Harbour Way on Regatta Boulevard at Seadrift Drive or to Hilltop Mall/Castro Ranch Road on Regatta Boulevard at Melville Square.

```
"InfoLinks": {
  "InfoLink": {
    "Uri": null
  }
},
"Consequences": {
  "Consequence": {
    "Severity": "normal",
    "Affects": {
      "Operators": {
        "AffectedOperator": {
          "OperatorRef": "AC Transit",
          "OperatorName": "AC"
        }
      },
      "Networks": {
        "AffectedNetwork": {
          "AffectedLine": {
            "LineRef": "74"
          }
        }
      }
    }
  }
}
```

B.1.9 Example Transit Scheduled Departures for a Stop Response (JSON) in SIRI ST format

```
{
  "Siri":{
    "ServiceDelivery":{
      "ResponseTimestamp": "2013-02-18T09:30:47-08:00",
      "Status":true,
      "StopTimetableDelivery":{
        "version":1.4,
        "ResponseTimestamp": "2004-12-17T09:30:47-05:00",
        "TimetabledStopVisit":{
          "RecordedAtTime": "2004-12-17T09:25:46-05:00",
          "MonitoringRef": "HLTST011",
          "TargetedVehicleJourney":{
            "LineRef":17,
            "DirectionRef": "INBOUND",
            "DatedVehicleJourneyRef": "TRP123214",
            "PublishedLineName": "Fremont",
            "OperatorRef": "BA",
            "OriginRef": "BART_11",
            "OriginName": "BART_CIVIC CENTER",
            "DestinationRef": "BART_99",
            "DestinationName": "BART_16th St-Mission",
          }
        }
      }
    }
  }
}
```
B.1.10 Example Transit Real Time Predictions at a StopResponse (JSON) in SIRI format

```json
{
   "Siri":{
      "ServiceDelivery":{
         "ResponseTimestamp": "2004-12-17T09:30:46-05:00",
         "ProducerRef": "BA",
         "Status":true,
         "StopMonitoringDelivery":{
            "version":1.4,
            "ResponseTimestamp": "2004-12-17T09:30:47-05:00",
            "Status":true,
            "MonitoredStopVisit":{
               "RecordedAtTime": "2004-12-17T09:25:46-05:00",
               "MonitoringRef": "EMBR",
               "MonitoredVehicleJourney":{
                  "LineRef": "Warm Springs/South Fremont - Daly City",
                  "DirectionRef": "E",
                  "FramedVehicleJourneyRef":{
                     "DataFrameRef": "2004-12-17",
                     "DatedVehicleJourneyRef": "1031357WKDY"
                  }
               }
            }
         }
      }
   }
}
```
"PublishedLineName": "Warm Springs/South Fremont - Daly City",
"OperatorRef": "BA",
"OriginRef": "DALY",
"OriginName": "Daly City BART Station",
"DestinationRef": "WARM",
"DestinationName": "Warm Springs/South Fremont",
"Monitored": true,
"InCongestion": false,
"VehicleLocation": {
  "Longitude": 180,
  "Latitude": 90
},
"ProgressStatus": "Service running on time",
"Bearing": 23,
"Occupancy": "full",
"VehicleRef": "1011",
"PreviousCalls": {
  "PreviousCall": {
    "StopPointRef": "BART_10",
    "VisitNumber": 2,
    "StopPointName": "BART_DALY CITY",
    "VehicleAtStop": false,
    "AimedDepartureTime": "2004-12-17T09:32:43-05:00",
    "ActualDepartureTime": "2004-12-17T09:32:43-05:00"
  }
},
"MonitoredCall": {
  "StopPointRef": "EMBR",
  "VisitNumber": 1,
  "StopPointName": "Embarcadero BART Station",
  "VehicleAtStop": false,
  "VehicleLocationAtStop": {
    "Longitude": 180,
    "Latitude": 90
  },
  "AimedArrivalTime": "2004-12-17T09:40:46-05:00",
  "ExpectedArrivalTime": "2004-12-17T09:40:46-05:00",
  "AimedDepartureTime": "2004-12-17T09:42:47-05:00",
  "ExpectedDepartureTime": "2004-12-17T09:40:47-05:00"
},
"OnwardCalls": {
  "OnwardCall": {
    "StopPointRef": "BART_12",
    "VisitNumber": 4,
    "StopPointName": "BAR_12th St Oakland",
    "VehicleAtStop": false,
    "AimedArrivalTime": "2004-12-17T09:30:56-05:00",
    "ExpectedArrivalTime": "2004-12-17T09:30:56-05:00",
    "AimedDepartureTime": "2004-12-17T09:30:57-05:00",
    "ExpectedDepartureTime": "2004-12-17T09:30:57-05:00"
  }
}
B.1.1.1 Example Real Time Vehicle Monitoring Response (JSON) in SIRI format

```json
{
  "Siri": {
    "ServiceDelivery": {
      "ResponseTimestamp": "2004-12-17T09:30:47-05:00",
      "ProducerRef": "BA",
      "Status": true,
      "VehicleMonitoringDelivery": {
        "version": 1.4,
        "ResponseTimestamp": "2004-12-17T09:30:47-05:00",
        "VehicleActivity": {
          "RecordedAtTime": "2004-12-17T09:30:47-05:00",
          "ValidUntilTime": "2004-12-17T09:30:47-05:00",
          "MonitoredVehicleJourney": {
            "LineRef": "17",
            "DirectionRef": "OB",
            "FramedVehicleJourneyRef": {
              "DataFrameRef": "2004-12-17"
            }
          }
        }
      }
    }
  }
}
```
"DatedVehicleJourneyRef":987675,
"PublishedLineName":123,
"OperatorRef": "BA",
"OriginName": "SFO",
"Via": [
  {
    "PlaceName": "16th st"
  },
  {
    "PlaceName": "West Oakland"
  }
],
"DestinationRef": "Fremont",
"DestinationName": "Fremont",
"Monitored": true,
"InCongestion": false,
"VehicleLocation": {
  "Longitude": 180,
  "Latitude": 90
},
"Bearing": 123,
"Occupancy": "full",
"ProgressRate": "slowProgress",
"Delay": "PT2M",
"ProgressStatus": "On time",
"VehicleRef": "VEH987654",
"PreviousCalls": {
  "PreviousCall": {
    "StopPointRef": "SFO",
    "VisitNumber": 2,
    "StopPointName": "String",
    "VehicleAtStop": false,
    "AimedDepartureTime": "2004-12-17T09:32:43-05:00",
    "ActualDepartureTime": "2004-12-17T09:32:43-05:00"
  }
},
"OnwardCalls": {
  "OnwardCall": {
    "StopPointRef": "80",
    "VisitNumber": 4,
    "StopPointName": "16th Street",
    "VehicleAtStop": false,
    "AimedArrivalTime": "2004-12-17T09:30:56-05:00",
    "ExpectedArrivalTime": "2004-12-17T09:30:56-05:00",
    "AimedDepartureTime": "2004-12-17T09:30:57-05:00",
    "ExpectedDepartureTime": "2004-12-17T09:30:57-05:00"
  }
}
},
"RecordedAtTime": "2004-12-17T09:30:47-05:00",
"ValidUntilTime": "2004-12-17T09:30:47-05:00",
"VehicleMonitoringRef": 45678,
"MonitoredVehicleJourney": {
  "LineRef": "Line123",
  "FramedVehicleJourneyRef": {
    "DataFrameRef": "2004-12-17"
"DatedVehicleJourneyRef": "Outbound",
"Monitored":true,
"VehicleLocation":{
  "Longitude":180,
  "Latitude":90
},
"Delay": "PT2M",
"VehicleRef": "VEH987654",
"OnwardCalls":{
  "OnwardCall":{
    "StopPointRef": "HLTST012",
    "StopPointName": "Church"
  }
}
"VehicleActivityCancellation":{
  "RecordedAtTime": "2004-12-17T09:30:47-05:00",
  "VehicleMonitoringRef":9876542,
  "VehicleJourneyRef":{
    "DataFrameRef": "2001-12-17",
    "DatedVehicleJourneyRef": "09867"
  },
  "LineRef": "Line123",
  "DirectionRef": "OB",
  "Reason": "Done for the day"
}

B.1.12 Example Transit Schedule Update Response (JSON) in SIRI PT format
{
  "Siri":{
    "ServiceDelivery":{
      "ResponseTimestamp": "2013-02-18T09:30:47-08:00",
      "Status":true,
      "ProductionTimetableDelivery":{
        "version":1.4,
        "ResponseTimestamp": "2004-12-17T09:30:47-05:00",
        "ValidUntil": "2001-12-17T10:30:47-05:00",
        "DatedTimetableVersionFrame":{
          "RecordedAtTime": "2001-12-17T09:30:47-05:00",
          "LineRef":123,
          "DirectionRef": "Out",
          "PublishedLineName": "String",
          "DatedVehicleJourney":{
            "DatedVehicleJourneyCode": "DVC0008767",
            "DatedCalls":{
              "DatedCall":{
                "StopPointRef": "BART_11",
                "CallNote": "optional message here",
```
"AimedArrivalTime": "2013-02-19T09:55:47-08:00",
"AimedDepartureTime": "2013-02-19T09:56:47-08:00"
},
{"StopPointRef": "BART_99",
"CallNote": "optional message here",
"AimedArrivalTime": "2013-02-19T10:15:47-08:00",
"AimedDepartureTime": "2013-02-19T10:16:47-08:00"
}

B.1.13 Example Transit Addition and Cancellation of Trip Response (JSON) in SIRI ET format

```json
{
  "Siri": {
    "xmlns": "http://www.siri.org.uk/siri",
    "xmlns:xsi": "http://www.w3.org/2001/XMLSchema-instance",
    "version": "1.4",
    "ServiceDelivery": {
      "ResponseTimestamp": "2013-02-18T09:30:47-08:00",
      "Status": true,
      "EstimatedTimetableDelivery": {
        "version": "1.4",
        "ResponseTimestamp": "2004-12-17T09:30:47-05:00",
        "EstimatedJourneyVersionFrame": {
          "RecordedAtTime": "2013-02-18T09:30:47-08:00",
          "EstimatedVehicleJourney": {
            "LineRef": "917",
            "DirectionRef": "INBOUND",
            "DatedVehicleJourneyRef": "00008",
            "PublishedLineName": "Fremont",
            "EstimatedCalls": {
              "EstimatedCall": {
                "StopPointRef": "BART_11",
                "CallNote": "optional message here",
                "AimedArrivalTime": "2013-02-18T09:55:47-08:00",
                "AimedDepartureTime": "2013-02-18T09:56:47-08:00"
              },
              "StopPointRef": "BART_99",
              "CallNote": "optional message here",
              "AimedArrivalTime": "2013-02-18T10:15:47-08:00",
              "AimedDepartureTime": "2013-02-18T10:16:47-08:00"
            }
          }
        }
      }
    }
  }
}
```
"LineRef":764,
"DirectionRef": "INBOUND",
"DatedVehicleJourneyRef": "00008",
"Cancellation":true,
"PublishedLineName": "Pittsburgh Bay Point"
}
}
}
}

B.1.14 Example Transit General Messaging Service Response (JSON) in SIRI GM format

```json
{
  "Siri":{
    "ResponseTimestamp": "2013-02-17T09:30:46-08:00",
    "Status":true,
    "GeneralMessageDelivery":{
      "version":1.4,
      "ResponseTimestamp": "2001-12-17T09:30:47.0Z",
      "GeneralMessage":{
        "formatRef": "string",
        "RecordedAtTime": "2013-02-17T09:30:46-08:00",
        "InfoMessageIdentifier": 12345,
        "InfoMessageVersion": 2,
        "InfoChannelRef": "WARNINGS",
        "ValidUntilTime": "2013-02-18T09:30:46-08:00",
        "Content": "some message here"
      },
        "formatRef": "string",
        "RecordedAtTime": "2013-02-17T09:30:46-08:00",
        "InfoMessageIdentifier": 23456,
        "InfoMessageVersion": 1,
        "InfoChannelRef": "WARNINGS",
        "ValidUntilTime": "2013-02-18T09:30:46-08:00",
        "Content": "some message here"
      }
    }
  }
}
```
B.1.15 Example GTFS Operator List in JSON format

```json
[
    {
        "Id": "3D",
        "Name": "Tri Delta Transit",
        "LastGenerated": "9/8/2017 5:22:04 PM"
    },
    {
        "Id": "AC",
        "Name": "AC Transit",
        "LastGenerated": "8/27/2017 6:06:13 PM"
    },
    {
        "Id": "AM",
        "Name": "Capitol Corridor Joint Powers Authority",
        "LastGenerated": "8/22/2017 11:23:13 AM"
    },
    {
        "Id": "AT",
        "Name": "Angel Island Tiburon Ferry",
        "LastGenerated": "9/13/2017 12:07:06 PM"
    },
    {
        "Id": "AY",
        "Name": "American Canyon Transit",
        "LastGenerated": "9/13/2017 12:02:18 PM"
    },
    {
        "Id": "RG",
        "Name": "Regional GTFS",
        "LastGenerated": "6/15/2020 5:11:56 AM"
    },
    {
        "Id": "BA",
        "Name": "BART",
        "LastGenerated": "6/6/2017 1:26:30 PM"
    }
]
```

B.1.16 Example Transit ServiceAlerts Response in JSON format

```json
{
    "Header": {
        "GtfsRealtimeVersion": "1.0",
        "incrementality": 0,
        "Timestamp": 1590704336
    },
    "Entities": [
        {
            "Id": "3469",
            "TripUpdate": null,
            "Vehicle": null,
            "Alert": {
                "ActivePeriods": [
                    {
                    
                    }
                ]
            }
        }
    ]
}
```
"Start": 1532736000,
"End": 1609488000
],
"InformedEntities": [
{
"AgencyId": "5S",
"Trip": null
}
],
"cause": 1,
"effect": 8,
"Url": {
"Translations": [
{
"Text": "",
"Language": "en"
}
],
"HeaderText": {
"Translations": [
{
"Text": "Call 511 or visit 511.org for more alert information. Issues with this sign? feedback@511.org",
"Language": "en"
}
],
"DescriptionText": {
"Translations": [
{
"Text": "Call or visit 511.org for more real-time departures and alert information."
,"Language": "en"
}
],
"TtsHeaderText": null,
"TtsDescriptionText": null
}
],
"Id": "21538899",
"TripUpdate": null,
"Vehicle": null,
"Alert": {
"ActivePeriods": [
{
"Start": 1583089140,
"End": 1704049140
}
],
"InformedEntities": [
{
"AgencyId": "5S",
"Trip": null
}
B.1.17 Example Transit Shapes Response in JSON format

```json
{
  "Content": {
    "TimetableFrame": {
      "version": "any",
      "id": "TF:BA",
      "vehicleJourneys": {
        "ServiceJourney": {
          "version": "any",
          "id": "3010811SUN",
          "LinkSequenceProjection": {
            "version": "any",
            "id": "245016",
            "LineString": {
              "srsName": "WGS84",
              "id": "245016",
              "pos": [
                "37.60808289 -122.3951075",
                ...
```
"37.60939749 -122.3964303",
"37.61078442 -122.3978387",
"37.61178447 -122.3988195",
"37.6124882 -122.3992295",
"37.61264766 -122.3994088",
"37.61288218 -122.3994895",
"37.61297159 -122.399568",
"37.61317171 -122.3996879",
"37.61334305 -122.3997649",
"37.61351504 -122.3998161",
"37.61380724 -122.3998666",
"37.61393581 -122.3998685",
"37.61423276 -122.3998292",
"37.61441157 -122.3997582",
"37.6146182 -122.3996593",
"37.61478632 -122.3995339",
"37.61494871 -122.3993958",
"37.61500888 -122.3993223",
"37.61513029 -122.399157",
"37.61526027 -122.3989516",
"37.61533452 -122.3988037",
"37.61541787 -122.3985962",
"37.61550791 -122.3982678",
"37.61551978 -122.3981671",
"37.61556776 -122.3976527",
"37.61557464 -122.3974131",
"37.61559137 -122.3969052",
"37.61558998 -122.3961968",
"37.61560202 -122.3948619",
"37.6156106 -122.3947422",
"37.61564872 -122.3942105",
"37.61602033 -122.3926008"
5 Appendix C: API Data Structures

5.1 SIRI

C.1.8 Announcement Message Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreationTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Time of the creation of the situation.</td>
</tr>
<tr>
<td>SituationNumber</td>
<td>Integer</td>
<td>Mandatory</td>
<td>Unique identifier for the situation.</td>
</tr>
<tr>
<td>Source</td>
<td>Container</td>
<td>Mandatory</td>
<td>Information about source of information</td>
</tr>
<tr>
<td>—SourceType</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Nature of source (feed, email, text, etc.)</td>
</tr>
<tr>
<td>—Name</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of source</td>
</tr>
<tr>
<td>ValidityPeriod</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is a container for validity period of the situation</td>
</tr>
<tr>
<td>—StartTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>It is inclusive start time of the situation</td>
</tr>
<tr>
<td>—EndTime</td>
<td>DateTime</td>
<td>Optional</td>
<td>It is inclusive end time stamp for situation. If omitted the situation is interpreted as to be forever.</td>
</tr>
<tr>
<td>Priority</td>
<td>Non Negative Integer</td>
<td>Optional</td>
<td>An arbitrary rating of the situation priority (1=high).</td>
</tr>
<tr>
<td>ScopeType</td>
<td>Enum</td>
<td>Optional</td>
<td>Provides the nature of scope, e.g. general, network etc.</td>
</tr>
<tr>
<td>Summary</td>
<td>Free Text</td>
<td>Optional</td>
<td>It is the summary of situation, id absent it is derived from situation Description</td>
</tr>
<tr>
<td>Description</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Description of the situation</td>
</tr>
<tr>
<td>InfoLinks</td>
<td>Container</td>
<td>Optional</td>
<td>Hyperlinks to other resources associated with situation</td>
</tr>
<tr>
<td>—InfoLink</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is container for the hyperlink associated with situation</td>
</tr>
<tr>
<td>—Uri</td>
<td>Link</td>
<td>Mandatory</td>
<td>Hyperlink associated with situation</td>
</tr>
<tr>
<td>Consequences</td>
<td>Container</td>
<td>Mandatory</td>
<td>It is the collection of consequence (SIRI element) which describes effect of the situation on Public Transport system. It has at least one consequence</td>
</tr>
</tbody>
</table>

**Consequence structure**

The Consequence structure is the main element of the Consequences collection. It contains information about the nature of the effect or disrupt on to the public transport service.
### C.1.9 Transit Scheduled Departures for a Stop Message Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of server response.</td>
</tr>
<tr>
<td>Status</td>
<td>Enum</td>
<td>Optional</td>
<td>Indicates success or failure of request.</td>
</tr>
<tr>
<td>true - success</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>false - failure</td>
<td></td>
<td></td>
<td>SIRI error response will be returned.</td>
</tr>
<tr>
<td>StopTimetableDelivery</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains multiple TimetabledStopVisit nodes, one for each visit to the stop within the Departure window.</td>
</tr>
</tbody>
</table>

**StopTimetableDelivery structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of server response.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Mandatory / Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------</td>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SubscriptionRef</td>
<td>Xsd:NMToken</td>
<td>Mandatory</td>
<td>Identifier of service subscription- unique within Service and Subscriber</td>
</tr>
<tr>
<td>TimetabledStopVisit</td>
<td>Object</td>
<td>Mandatory</td>
<td>A visit to a stop by a vehicle in the production timetable</td>
</tr>
</tbody>
</table>

**TimetabledStopVisit structure**

This contains details on a single visit to the stop within the Departure window.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>Date/Time</td>
<td>Mandatory</td>
<td>Date and time when data was recorded.</td>
</tr>
<tr>
<td>MonitoringRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Identifier of stop monitoring point that Stop Visit applies.</td>
</tr>
<tr>
<td>TargetedVehicleJourney</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains a single TargetedVehicleJourney node.</td>
</tr>
</tbody>
</table>

**TargetedVehicleJourney structure**

This contains details on a single visit to the stop within the Departure window.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>A Line in SIRI is equivalent to a Route in GTFS. Value is RouteCode e.g.: &quot;917&quot; = &quot;Fremont&quot; for &quot;BA&quot; agency.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Value could be either INBOUND or OUTBOUND etc.</td>
</tr>
<tr>
<td>FramedVehicleJourneyRef</td>
<td>Object</td>
<td>Optional</td>
<td>A compound element uniquely identifying the trip the vehicle is serving.</td>
</tr>
<tr>
<td>PublishedLineName</td>
<td>Free Text</td>
<td>Optional</td>
<td>Value is Route Name e.g.: &quot;Fremont&quot; for &quot;BA&quot; agency.</td>
</tr>
<tr>
<td>OperatorRef</td>
<td>Reference ID</td>
<td>Optional</td>
<td>Operator of the journey</td>
</tr>
<tr>
<td>OriginRef</td>
<td>Computed Text</td>
<td>Optional</td>
<td>The stop ID for the first stop on the trip the vehicle is serving, prefixed by Agency Name and or Route Name to make it unique e.g.: &quot;BART_11&quot;.</td>
</tr>
</tbody>
</table>
### OriginName
- **Type**: Free Text
- **Mandatory/Optional**: Optional
- **Description**: The stop Name for the first stop on the trip the vehicle is serving, prefixed by Agency Name e.g.: “BART_CIVIC CENTER”.

### DestinationRef
- **Type**: Computed Text
- **Mandatory/Optional**: Optional
- **Description**: The stop ID for the last stop on the trip the vehicle is serving, prefixed by Agency Name e.g.: “BART_99”.

### DestinationName
- **Type**: Free Text
- **Mandatory/Optional**: Optional
- **Description**: The stop Name for the last stop on the trip the vehicle is serving, prefixed by Agency Name e.g.: “BART_16th St-Mission”.

### VehicleJourneyName
- **Type**: Free Text
- **Mandatory/Optional**: Optional
- **Description**: The trip headsign corresponding to the trip (journey) the vehicle is serving.

### TargetedCall
- **Type**: Object
- **Mandatory/Optional**: Optional
- **Description**: Contains a single TargetedCall node.

#### FramedVehicleJourneyRef Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataFrameRef</td>
<td>Date time</td>
<td>Mandatory</td>
<td>The service date for the trip the vehicle is serving.</td>
</tr>
<tr>
<td>DatedVehicleJourneyRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>The trip ID for trip the vehicle is serving.</td>
</tr>
</tbody>
</table>

#### TargetedCall structure

This describes the arrival and departure times for a specific visit.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VisitNumber</td>
<td>Numeric</td>
<td>Mandatory</td>
<td>For journey patterns that involve repeated visits by a vehicle to a stop, the VisitNumber count is used to distinguish each separate visit.</td>
</tr>
<tr>
<td>AimedArrivalTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Value is expected arrival time.</td>
</tr>
<tr>
<td>AimedDepartureTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Value is expected departure time.</td>
</tr>
</tbody>
</table>

#### C.1.10 Real-time predictions at a Stop Message Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of response from server.</td>
</tr>
</tbody>
</table>
### Status

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Indicates success or failure of request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>true - success</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>false - failure, SIRI error response will be returned</td>
</tr>
</tbody>
</table>

### StopMonitoringDelivery

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StopMonitoringDelivery</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains multiple MonitoredStopVisit entries, one per visit to the stop.</td>
</tr>
</tbody>
</table>

### StopMonitoringDelivery structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MonitoredStopVisit</td>
<td>Object</td>
<td>Required</td>
<td>This contains monitored vehicle journey (real-time trip) information.</td>
</tr>
<tr>
<td>MonitoredStopVisitCancellation</td>
<td>Object</td>
<td>Optional</td>
<td>This contains cancellation information for a trip.</td>
</tr>
<tr>
<td>StopLineNotice</td>
<td>Object</td>
<td>Optional</td>
<td>This provides notices for lines serving this monitored stop.</td>
</tr>
<tr>
<td>StopLineNoticeCancellation</td>
<td>Object</td>
<td>Optional</td>
<td>This provides cancellation of previous issued notices for lines serving this monitored stop.</td>
</tr>
</tbody>
</table>

### MonitoredStopVisit structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Required</td>
<td>The timestamp of the last real-time update from the particular vehicle.</td>
</tr>
<tr>
<td>MonitoringRef</td>
<td>Free Text</td>
<td>Optional</td>
<td>Name of the Stop being monitored</td>
</tr>
<tr>
<td>MonitoredVehicleJourney</td>
<td>Object</td>
<td>Optional</td>
<td>Real-time information about particular vehicles</td>
</tr>
</tbody>
</table>

### MonitoredVehicleJourney structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OperatorRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For AgencyCode requirement, e.g.: “BA”.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Could be moved under sub-node Extensions because it’s NOT part of the SIRI spec.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Mandatory</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------</td>
<td>-----------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For Route Code requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A Line in SIRI is equivalent to a Route in GTFS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Value could either be RouteCode or RouteName as required. Recommend using RouteCode because &quot;PublishedLineName&quot; is using RouteName.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: RouteCode &quot;917&quot; = RouteName &quot;Fremont&quot; for BART.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Does not identify the Agency, so RouteCode or RouteName would have to be unique to an Agency.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Defined Text</td>
<td>Mandatory</td>
<td>For Direction requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>E.g.: &quot;IB&quot;</td>
</tr>
<tr>
<td>FramedVehicleJourneyRef</td>
<td>Object</td>
<td>Mandatory</td>
<td>A compound element uniquely identifying the trip the vehicle is serving.</td>
</tr>
<tr>
<td>PublishedLineName</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For Route name requirement.</td>
</tr>
<tr>
<td>OriginRef</td>
<td>Computed Text</td>
<td>Optional</td>
<td>&quot;The GTFS stop code or stop ID for the first stop on the trip the vehicle is serving.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;DALY&quot;</td>
</tr>
<tr>
<td>OriginName</td>
<td>Free Text</td>
<td>Optional</td>
<td>For Origin place name requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&quot;The GTFS stop Name for the first stop on the trip the vehicle is serving.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;Daly City BART Station&quot;</td>
</tr>
<tr>
<td>DestinationRef</td>
<td>Computed Text</td>
<td>Optional</td>
<td>&quot;The GTFS stop code or stop ID for the last stop on the trip the vehicle is serving, prefixed by Agency ID.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;WARM&quot;</td>
</tr>
<tr>
<td>DestinationName</td>
<td>Free Text</td>
<td>Optional</td>
<td>For Destination place name requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&quot;The trip head sign, if available. If not, GTFS stop Name for the last stop on the trip the vehicle is serving.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;Warm Springs/South Fremont&quot;</td>
</tr>
<tr>
<td>Monitored</td>
<td>Boolean</td>
<td>Mandatory</td>
<td>True if the trip is monitored.</td>
</tr>
<tr>
<td>InCongestion</td>
<td>Boolean</td>
<td>Optional</td>
<td>Indicates the congestion level affecting the vehicle. Set to true if the agency provides the congestion</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Mandatory / Optional</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------</td>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Bearing</td>
<td>Float</td>
<td>Optional</td>
<td>When data is available, this field provides values in degrees, clockwise from True North, i.e., 0 is North and 90 is East.</td>
</tr>
<tr>
<td>Occupancy</td>
<td>Enum</td>
<td>Optional</td>
<td>When data is available this field provides level of passenger occupancy in the vehicle - full, seatsAvailable, standingAvailable</td>
</tr>
<tr>
<td>VehicleRef</td>
<td>Free Text</td>
<td>Optional</td>
<td>Vehicle Identifier. Internal system identification of the vehicle. Should be unique per vehicle (for a given operator) and is used for tracking the vehicle as it proceeds through the system.</td>
</tr>
<tr>
<td>MonitoredCall</td>
<td>Object</td>
<td>Mandatory</td>
<td>Call data for the stop</td>
</tr>
<tr>
<td>OnwardsCalls</td>
<td>Object</td>
<td>Optional</td>
<td>Call data for next stops</td>
</tr>
<tr>
<td>PreviousCalls</td>
<td>Object</td>
<td>Optional</td>
<td>Call data for previous stops</td>
</tr>
<tr>
<td>ProgressStatus</td>
<td>Enum</td>
<td>Optional</td>
<td>Status of the current vehicle, On-time, Running early etc.</td>
</tr>
<tr>
<td>VehicleLocation</td>
<td>Object</td>
<td>Optional</td>
<td>Vehicle location information. (Latitude/Longitude)</td>
</tr>
</tbody>
</table>

**FramedVehicleJourneyRef Structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataFrameRef</td>
<td>Date time</td>
<td>Mandatory</td>
<td>The service date for the trip the vehicle is serving.</td>
</tr>
<tr>
<td>DatedVehicleJourneyRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>The trip ID for trip the vehicle is serving.</td>
</tr>
</tbody>
</table>

**Monitored/Onward/Previous Call structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StopPointRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Regional stop identifier of the stop that is being monitored.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Mandatory/Optional</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>StopPointName</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Name of the stop</td>
</tr>
<tr>
<td>VehicleLocationAtStop</td>
<td>Object</td>
<td>Optional</td>
<td>Vehicle location information at stop. (Latitude/Longitude)</td>
</tr>
<tr>
<td>VehicleAtStop</td>
<td>Boolean</td>
<td>Mandatory</td>
<td>True if vehicle is at the stop.</td>
</tr>
<tr>
<td>AimedArrivalTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Scheduled arrival time requirement.</td>
</tr>
<tr>
<td>ExpectedArrivalTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Predicted arrival time requirement.</td>
</tr>
<tr>
<td>AimedDepartureTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Scheduled departure time requirement.</td>
</tr>
<tr>
<td>ExpectedDepartureTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Predicted departure time requirement.</td>
</tr>
<tr>
<td>Distances</td>
<td>Object</td>
<td>Optional</td>
<td>Extension to SIRI Call structure to incorporate distance and bearing information of vehicle from the stop.</td>
</tr>
</tbody>
</table>

**Distances structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallDistanceAlongRoute</td>
<td>Numeric</td>
<td>Optional</td>
<td>Distance of the stop from the beginning of the trip/route</td>
</tr>
<tr>
<td>DistanceFromCall</td>
<td>Numeric</td>
<td>Optional</td>
<td>Distance from the vehicle to the stop along the route, in meters</td>
</tr>
<tr>
<td>StopsFromCall</td>
<td>Numeric</td>
<td>Optional</td>
<td>The number of stops on the vehicle’s current trip until the stop in question, starting from 0.</td>
</tr>
<tr>
<td>PresentableDistance</td>
<td>Text</td>
<td>Optional</td>
<td>Suggested display for the distance of vehicle from the stop.</td>
</tr>
</tbody>
</table>

**MonitoredStopVisitCancellation structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>The timestamp of the last real-time update from the particular vehicle.</td>
</tr>
<tr>
<td>MonitoringRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Name of the Stop being monitored.</td>
</tr>
</tbody>
</table>
**StopLineNotice structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>The timestamp of the last real-time update from the particular vehicle.</td>
</tr>
<tr>
<td>ItemRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Reference to a previously issued notice.</td>
</tr>
<tr>
<td>MonitoringRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Name of the Stop being monitored</td>
</tr>
<tr>
<td>LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For Route Code requirement. A Line in SIRI is equivalent to a Route in GTFS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Value could either be RouteCode or RouteName as required. Recommend using RouteCode because &quot;PublishedLineName&quot; is using RouteName.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: RouteCode &quot;917&quot; = RouteName &quot;Fremont&quot; for BART.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Does not identify the Agency, so RouteCode or RouteName would have to be unique to an Agency.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Defined Text</td>
<td>Mandatory</td>
<td>For Direction requirement. &quot;In&quot; = inbound, &quot;Out&quot; = outbound</td>
</tr>
<tr>
<td>Note</td>
<td>Free Text</td>
<td>Optional</td>
<td>Note about the cancellation.</td>
</tr>
</tbody>
</table>

**StopLineNoticeCancellation structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>The timestamp of the last real-time update from the particular vehicle.</td>
</tr>
<tr>
<td>ItemIdentifier</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Unique identifier for this notice</td>
</tr>
</tbody>
</table>
### MonitoringRef
- **Type**: Free Text
- **Mandatory**: Yes
- **Description**: Name of the Stop being monitored

#### LineRef
- **Type**: Free Text
- **Mandatory**: Yes

*For Route Code requirement.*

A Line in SIRI is equivalent to a Route in GTFS. Value could either be RouteCode or RouteName as required. Recommend using RouteCode because "PublishedLineName" is using RouteName.

e.g.: RouteCode "917" = RouteName "Fremont" for BART.

Does not identify the Agency, so RouteCode or RouteName would have to be unique to an Agency.

#### DirectionRef
- **Type**: Defined Text
- **Mandatory**: Yes

*For Direction requirement.*

"In" = inbound, "Out" = outbound

#### LineNote
- **Type**: Free Text
- **Mandatory**: Yes

Information about the notice.

### C.1.11 Real-time Vehicle Monitoring Message Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of response from server.</td>
</tr>
<tr>
<td>Status</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Indicates success or failure of request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>true - success</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>false - failure, SIRI error response will be returned</td>
</tr>
<tr>
<td>VehicleMonitoringDelivery</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains multiple <strong>VehicleActivity</strong> entries, one per trip, if monitored.</td>
</tr>
</tbody>
</table>

#### VehicleMonitoringDelivery structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Type</td>
<td>Mandatory/ Optional</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>VehicleActivity</td>
<td>Object</td>
<td>Required</td>
<td>This contains monitored vehicle journey (real-time trip) information.</td>
</tr>
<tr>
<td>VehicleActivityCancellation</td>
<td>Object</td>
<td>Optional</td>
<td>This contains cancellation information for a trip.</td>
</tr>
</tbody>
</table>

### VehicleActivity structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Required</td>
<td>The timestamp of the last real-time update from the particular vehicle.</td>
</tr>
<tr>
<td>ValidUntilTime</td>
<td>DateTime</td>
<td>Required</td>
<td>Time until which data is valid.</td>
</tr>
<tr>
<td>MonitoredVehicleJourney</td>
<td>Object</td>
<td>Optional</td>
<td>Real-time information about particular vehicles</td>
</tr>
</tbody>
</table>

### MonitoredVehicleJourney structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OperatorRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For AgencyCode requirement, e.g.: “BA”.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Could be moved under sub-node Extensions because it’s NOT part of the SIRI spec.</td>
</tr>
<tr>
<td>LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For Route Code requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A Line in SIRI is equivalent to a Route in GTFS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Value could either be RouteCode or RouteName as required. Recommend using RouteCode because &quot;PublishedLineName&quot; is using RouteName.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: RouteCode &quot;917&quot; = RouteName &quot;Fremont&quot; for BART.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Does not identify the Agency, so RouteCode or RouteName would have to be unique to an Agency.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Defined Text</td>
<td>Mandatory</td>
<td>For Direction requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>E.g.: “IB”</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------</td>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FramedVehicleJourneyRef</td>
<td>Object</td>
<td>Mandatory</td>
<td>A compound element uniquely identifying the trip the vehicle is serving.</td>
</tr>
<tr>
<td>PublishedLineName</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>For Route name requirement.</td>
</tr>
<tr>
<td>OriginRef</td>
<td>Computed Text</td>
<td>Optional</td>
<td>&quot;The GTFS stop code or stop ID for the first stop on the trip the vehicle is serving, prefixed by Agency ID.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;DALY&quot;</td>
</tr>
<tr>
<td>OriginName</td>
<td>Free Text</td>
<td>Optional</td>
<td>For Origin place name requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&quot;The GTFS stop Name for the first stop on the trip the vehicle is serving.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>We don’t have an Agency ID, so would use Agency Name e.g.: &quot;Daly City BART Station&quot;</td>
</tr>
<tr>
<td>DestinationRef</td>
<td>Computed Text</td>
<td>Optional</td>
<td>&quot;The GTFS stop code or stop ID for the last stop on the trip the vehicle is serving, prefixed by Agency ID.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;WARM&quot;</td>
</tr>
<tr>
<td>DestinationName</td>
<td>Free Text</td>
<td>Optional</td>
<td>For Destination place name requirement.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&quot;The trip head sign, if available. If not, GTFS stop Name for the last stop on the trip the vehicle is serving.&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>e.g.: &quot;Warm Springs/South Fremont&quot;</td>
</tr>
<tr>
<td>Monitored</td>
<td>Boolean</td>
<td>Mandatory</td>
<td>True if the trip is monitored for real-time updates.</td>
</tr>
<tr>
<td>InCongestion</td>
<td>Boolean</td>
<td>Optional</td>
<td>Indicates the congestion level affecting the vehicle. Set to true if the agency provides the congestion level as STOP_AND_GO, CONGESTION or SEVERE_CONGESTION. Set to false if agency provides the congestion level as RUNNING_SMOOTHLY</td>
</tr>
<tr>
<td>Bearing</td>
<td>Float</td>
<td>Optional</td>
<td>When data is available, this field provides values in degrees, clockwise from True North, i.e., 0 is North and 90 is East.</td>
</tr>
<tr>
<td>Occupancy</td>
<td>Enum</td>
<td>Optional</td>
<td>When data is available this field provides level of passenger occupancy in the vehicle - full, seatsAvailable, standingAvailable</td>
</tr>
<tr>
<td>MonitoredCall</td>
<td>Object</td>
<td>Optional</td>
<td>Call data for the current stop</td>
</tr>
<tr>
<td>OnwardsCalls</td>
<td>Object</td>
<td>Optional</td>
<td>Call data for next stops</td>
</tr>
<tr>
<td>PreviousCalls</td>
<td>Object</td>
<td>Optional</td>
<td>Call data for previous stops</td>
</tr>
<tr>
<td>ProgressStatus</td>
<td>Enum</td>
<td>Optional</td>
<td>Status of the current vehicle, On-time, Running early etc.</td>
</tr>
</tbody>
</table>
### VehicleRef

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VehicleRef</td>
<td>Free Text</td>
<td>Optional</td>
<td>The unique identifier of the vehicle to be monitored.</td>
</tr>
</tbody>
</table>

### VehicleLocation

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VehicleLocation</td>
<td>Object</td>
<td>Optional</td>
<td>Vehicle location information. (Latitude/Longitude)</td>
</tr>
</tbody>
</table>

### FramedVehicleJourneyRef Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataFrameRef</td>
<td>Date time</td>
<td>Mandatory</td>
<td>The service date for the trip the vehicle is serving.</td>
</tr>
<tr>
<td>DatedVehicleJourneyRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>The trip ID for trip the vehicle is serving.</td>
</tr>
</tbody>
</table>

### Monitored/Onward/Previous Call structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory / Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StopPointRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Stop code or stop ID.</td>
</tr>
<tr>
<td>StopPointName</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Name of the stop</td>
</tr>
<tr>
<td>VehicleLocationAtStop</td>
<td>Object</td>
<td>Optional</td>
<td>Vehicle location information at stop. (Latitude/Longitude)</td>
</tr>
<tr>
<td>VehicleAtStop</td>
<td>Boolean</td>
<td>Optional</td>
<td>True if vehicle is at the stop.</td>
</tr>
<tr>
<td>AimedArrivalTime</td>
<td>DateTime</td>
<td>Optional</td>
<td>Scheduled arrival time requirement.</td>
</tr>
<tr>
<td>ExpectedArrivalTime</td>
<td>DateTime</td>
<td>Optional</td>
<td>Predicted arrival time requirement.</td>
</tr>
<tr>
<td>ActualArrivalTime</td>
<td>Date Time</td>
<td>Optional</td>
<td>Observed arrival time.</td>
</tr>
<tr>
<td>AimedDepartureTime</td>
<td>DateTime</td>
<td>Optional</td>
<td>Scheduled departure time requirement.</td>
</tr>
<tr>
<td>ExpectedDepartureTime</td>
<td>DateTime</td>
<td>Optional</td>
<td>Predicted departure time requirement.</td>
</tr>
<tr>
<td>ActualDepartureTime</td>
<td>Date Time</td>
<td>Optional</td>
<td>Observed departure time.</td>
</tr>
<tr>
<td>Distances</td>
<td>Object</td>
<td>Optional</td>
<td>Extension to SIRI Call structure to incorporate distance and bearing information of vehicle from the stop.</td>
</tr>
</tbody>
</table>

### Distances structure
### Field | Type     | Mandatory/Optional | Description |
--- | --- | --- | --- |
CallDistanceAlongRoute | Numeric | Optional | Distance of the stop from the beginning of the trip/route |
DistanceFromCall | Numeric | Optional | Distance from the vehicle to the stop along the route, in meters |
StopsFromCall | Numeric | Optional | The number of stops on the vehicle's current trip until the stop in question, starting from 0. |
PresentableDistance | Text | Optional | Suggested display for the distance of vehicle from the stop. |

#### VehicleActivityCancellation structure

### Field | Type     | Mandatory/Optional | Description |
--- | --- | --- | --- |
RecordedAtTime | Date Time | Mandatory | The timestamp when data was recorded. |
VehicleJourneyRef | Object | Mandatory | A compound element uniquely identifying the trip the vehicle is serving. |
LineRef | Free Text | Mandatory | A Line in SIRI is equivalent to a Route in GTFS. Value could either be RouteCode or RouteName as required. Recommend using RouteCode because "PublishedLineName" is using RouteName. e.g.: RouteCode "917" = RouteName "Fremont" for BART. Does not identify the Agency, so RouteCode or RouteName would have to be unique to an Agency. |
DirectionRef | Defined Text | Mandatory | For Direction requirement. “In” =inbound, “Out” = outbound |
Reason | Free Text | Mandatory | Reason for cancellation of this trip. For e.g. Vehicle has completed all its journeys. |

### C.1.12 Transit Schedule Updates for an agency Message Structure
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of server response.</td>
</tr>
<tr>
<td>Status</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Indicates success or failure of request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>true - success</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>false - failure, SIRI error response will be returned</td>
</tr>
<tr>
<td>ProductionTimetableDelivery</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains multiple DatedTimetableVersionFrame nodes.</td>
</tr>
</tbody>
</table>

**ProductionTimetableDelivery structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of server response.</td>
</tr>
<tr>
<td>DatedTimetableVersionFrame</td>
<td>Object</td>
<td>Mandatory</td>
<td>A timetable to run on a specific date</td>
</tr>
</tbody>
</table>

**DatedTimetableVersionFrame structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Date and time when data was recorded.</td>
</tr>
<tr>
<td>LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>A Line in SIRI is equivalent to a Route in GTFS. Value is RouteCode e.g.: &quot;917&quot; = &quot;Fremont&quot; for &quot;BART&quot; agency.</td>
</tr>
<tr>
<td>DirectionRef</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Value is either INBOUND or OUTBOUND</td>
</tr>
<tr>
<td>PublishedLineName</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Value is Route Name e.g.: &quot;Fremont&quot; for &quot;BART&quot; agency.</td>
</tr>
<tr>
<td>LineNote</td>
<td>Free Text</td>
<td>Optional</td>
<td>Text message describing this change.</td>
</tr>
<tr>
<td>DatedVehicleJourney</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains a DatedVehicleJourney node.</td>
</tr>
</tbody>
</table>

**DatedVehicleJourney structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatedVehicleJourneyCode</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Identifies the vehicle journey (Tripid).</td>
</tr>
<tr>
<td>DatedCalls</td>
<td>Objects</td>
<td>Mandatory</td>
<td>May contain multiple DatedCall nodes.</td>
</tr>
</tbody>
</table>
DatedCall structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StopPointRef</td>
<td>Numeric</td>
<td>Mandatory</td>
<td>The GTFS stop ID for this stop on the trip the vehicle is serving, prefixed by Agency Name e.g.: &quot;BART_11&quot;.</td>
</tr>
<tr>
<td>AimedArrivalTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Value is expected arrival time.</td>
</tr>
<tr>
<td>AimedDepartureTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Value is expected departure time.</td>
</tr>
<tr>
<td>CallNote</td>
<td>Text</td>
<td>Optional</td>
<td>Text message describing this change.</td>
</tr>
</tbody>
</table>

C.1.13 Transit Addition and Cancellation of Trips by Agency Message Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of server response.</td>
</tr>
<tr>
<td>Status</td>
<td>Enum</td>
<td>Mandatory</td>
<td>Indicates success or failure of request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>true - success</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>false - failure, SIRI error response will be returned</td>
</tr>
<tr>
<td>EstimatedTimetableDelivery</td>
<td>Object</td>
<td>Mandatory</td>
<td>Contains multiple EstimatedJourneyVersionFrame node.</td>
</tr>
</tbody>
</table>

EstimatedJourneyVersionFrame structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Date and time when data was recorded.</td>
</tr>
<tr>
<td>EstimatedVehicleJourney</td>
<td>Object</td>
<td>Mandatory</td>
<td>May contain multiple EstimatedVehicleJourney nodes, one for each vehicle.</td>
</tr>
</tbody>
</table>

EstimatedVehicleJourney structure

Provides real-time information about a journey along which a vehicle is running.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LineRef</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>A Line in SIRI is equivalent to a Route in GTFS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Value is RouteCode e.g.: &quot;917&quot; = &quot;Fremont&quot; for “BART” agency.</td>
</tr>
</tbody>
</table>
### DirectionRef

**Type:** Enum  
**Mandatory:** Yes  
Value is either INBOUND or OUTBOUND

### DatedVehicleJourneyRef

**Type:** Free Text  
**Mandatory:** Yes  
Reference to a dated vehicle journey or trip.

### Cancellation

**Type:** Enum  
**Mandatory:** No  
Value is “true” if cancelled.

### PublishedLineName

**Type:** Free Text  
**Mandatory:** Yes  
Value is Route Name e.g.: “Fremont” for “BART” agency.

### EstimatedCalls

**Type:** Objects  
**Mandatory:** Yes  
May contain multiple `EstimatedCall` nodes. Not returned if journey is cancelled.

**EstimatedCall structure**

This describes the times at a stop. A journey must contain at least two calls.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StopPointRef</td>
<td>Numeric</td>
<td>Mandatory</td>
<td>The GTFS stop ID for this stop on the trip the vehicle is serving, prefixed by Agency Name e.g.: &quot;BART_11&quot;.</td>
</tr>
<tr>
<td>AimedArrivalTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Value is expected arrival time.</td>
</tr>
<tr>
<td>AimedDepartureTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Value is expected departure time.</td>
</tr>
<tr>
<td>CallNote</td>
<td>Text</td>
<td>Optional</td>
<td>Text message describing the update.</td>
</tr>
</tbody>
</table>

### C.1.14 General Announcements Message Structure

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Timestamp of server response.</td>
</tr>
</tbody>
</table>
| Status              | Enum         | Mandatory          | Indicates success or failure of request.  
false - failure, SIRI error response will be returned |
| GeneralMessageDelivery | Object      | Mandatory          | May contain multiple `GeneralMessage` nodes.                                |

**GeneralMessageDelivery structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResponseTimestamp</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Date and time when message was recorded.</td>
</tr>
<tr>
<td>GeneralMessage</td>
<td>Object</td>
<td>Optional</td>
<td>A message from an agency.</td>
</tr>
</tbody>
</table>
**GeneralMessage structure**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordedAtTime</td>
<td>DateTime</td>
<td>Mandatory</td>
<td>Date and time when message was recorded.</td>
</tr>
<tr>
<td>InfoMessageIdentifier</td>
<td>String</td>
<td>Optional</td>
<td>Unique identifier of this message.</td>
</tr>
<tr>
<td>InfoMessageVersion</td>
<td>Int</td>
<td>Optional</td>
<td>Version number of this message.</td>
</tr>
<tr>
<td>InfoChannelRef</td>
<td>Text</td>
<td>Optional</td>
<td>Information channel to which message belongs.</td>
</tr>
<tr>
<td>ValidUntilTime</td>
<td>DateTime</td>
<td>Optional</td>
<td>Date and time of message expiration. If not provided, message is open-ended.</td>
</tr>
<tr>
<td>Content</td>
<td>Free Text</td>
<td>Mandatory</td>
<td>Text message.</td>
</tr>
</tbody>
</table>

**C.1.15 ServiceAlerts Structure**

Described in the Google documentation at:

https://developers.google.com/transit/gtfs-realtime/service-alerts

https://developers.google.com/transit/gtfs-realtime/examples/alerts
6 Appendix D: GTFS+ Files Structures

D.1.1 directions.txt File Structure
This file contains descriptions for each of the direction_ids provided for a route in the GTFS trips.txt file.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route_id</td>
<td>Text</td>
<td>Mandatory</td>
<td>From GTFS routes.txt file.</td>
</tr>
<tr>
<td>Direction_id</td>
<td>Numeric</td>
<td>Mandatory</td>
<td>Binary direction_id from GTFS trips.txt file. Each (route_id, direction_id) pair is unique in directions.txt.</td>
</tr>
<tr>
<td>Direction</td>
<td>Text</td>
<td>Mandatory</td>
<td>Corresponding direction name. Following are the values for direction: North, South, East, West, Northeast, Northwest, Southeast, Southwest, Clockwise, Counterclockwise, Inbound, Outbound, Loop, A Loop, B Loop</td>
</tr>
</tbody>
</table>

D.1.2 calendar_attributes.txt File Structure
This file contains descriptions for each of the service_ids provided in the GTFS calendar.txt file.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service_id</td>
<td>Text</td>
<td>Mandatory</td>
<td>From GTFS calendar.txt file.</td>
</tr>
<tr>
<td>Service_description</td>
<td>Text</td>
<td>Mandatory</td>
<td>Description of the service. For example, Weekdays, Sunday/Holiday, etc.</td>
</tr>
</tbody>
</table>

D.1.3 farezone_attributes.txt File Structure
This file contains zone names for each of the zone_ids provided in the GTFS stops.txt file.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zone_id</td>
<td>Text</td>
<td>Mandatory</td>
<td>From GTFS stops.txt file.</td>
</tr>
</tbody>
</table>
D.1.4 rider_categories.txt File Structure

The GTFS fare_attributes.txt file provides the fares for the regular adult fare category only. This file lists the other rider categories that the agency may define for discounted fares.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rider_category_id</td>
<td>Numeric</td>
<td>Mandatory</td>
<td>Unique rider category ID. Values are: 2 - Senior, 3 - Child, 4 - Student, 5 - Youth, 6 - Disabled, 7 - Promotional category, 11 - Military, 15 to 25 - Custom categories defined by the agency.</td>
</tr>
<tr>
<td>Rider_category_description</td>
<td>Text</td>
<td>Mandatory</td>
<td>Rider category description, such as Child (ages 5-11), Seniors (Ages 62 &amp; Up).</td>
</tr>
</tbody>
</table>

D.1.5 fare_rider_categories.txt File Structure

GTFS file fare_attributes.txt contains the fares for the regular adult rider category. Fares for other rider categories defined in the rider_categories.txt file above such as Child, Senior, etc will be provided in this GTFS+ file. The combination of fare_id and rider_category_id will be unique in this file.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Mandatory/ Optional</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fare_id</td>
<td>Text</td>
<td>Mandatory</td>
<td>From GTFS fare_attributes.txt file.</td>
</tr>
<tr>
<td>Rider_category_id</td>
<td>Numeric</td>
<td>Mandatory</td>
<td>From GTFS+ file rider_categories.txt.</td>
</tr>
<tr>
<td>Price</td>
<td>Decimal</td>
<td>Mandatory</td>
<td>Fare for the specified fare_id and rider_category_id in USD.</td>
</tr>
</tbody>
</table>
Appendix E: Historic Regional GTFS Feed

The Historical Regional Feed is produced monthly and gives a complete, retrospective view of service scheduled on each day of the month. By combining multiple Historical Regional Feed months, users can perform analyses across months, quarters, and years.

Historical Regional Feed products are fully valid GTFS feeds, but they differ somewhat in their contents from the daily Regional Feed products. This document describes the process used to produce the Historical Regional Feed products and their key differences.

E.1.1 Slicing regional feeds

Each day, the Regional Feed is produced from the versions of agency feeds on 511.org that provide the best view of service on that day. Each month, the Historical Regional Feed creation process takes these Regional Feeds and combines them together, taking one day of service from each feed, which we are calling a “slice.”

For example:

<table>
<thead>
<tr>
<th>Feed filename</th>
<th>Published</th>
<th>Contributes service slice for</th>
</tr>
</thead>
</table>

If the Regional Feed for a given day is missing, the closest previous day provides service. For instance, if 2020-04-22 was missing, the 2020-04-21 feed slice would cover both 2020-04-21 and 2020-04-22.

E.1.2 Global entity copying

Agencies, stops, and routes are considered “global”, and are handled using a simple ID-based merge with the most recent version winning. For example, if BART has a route with ID “OR-S” that is called “Richmond - Warm Springs”, but then later renames it to “Richmond to Warm Springs”, then the latter version will be used.

E.1.3 Trip hashing, comparison, and copying

Trips are more complicated and handled separately. A simple combining of all the trips and stop_times in all of the input files can easily create a GTFS feed that is too large for practical use, especially given that programs like OpenTripPlanner need to hold the entire schedule in memory. Therefore, duplicate copies of trips are detected using a hash based approach and only copied to the output once. This reduces the output size by approximately 90%.

For example, here are three hypothetical versions of Trip ID “BA:2210503” from three consecutive days of input regional feeds.
For each of these, the hashing function takes into account all trip attributes, all the calendar attributes for that trip, and the full details of each entry in stop_times.txt. Any change in any field will result in a different hash. This allows us to directly compare trips between versions of the input feed. Above, all details and schedule for 2020-04-23 and 2020-04-24 match exactly, so these trips two will be considered identical. The trip for 2020-04-22 has some minor differences in name and schedule, so will generate a different hash, and be considered a different trip.

As the historical feed merging program processes each input feed, it calculates the hash of each trip in the feed. If it has not seen a trip before, it copies it to the output and notes the hash for future use. If it has been seen before, it is not copied again. To prevent clashes, the original Trip IDs are appended with the trip hash (e.g. BA:2210503 -> BA:2210503:8c4ecb). The merging program then takes all trips in the input feed (both seen and unseen) and examines the calendars to see which are active for each day in this slice, and then creates calendar_dates.txt entries for each trip on each day where that trip is scheduled to run. The original service IDs are changed to be the same as the hash appended Trip ID, and the calendar is unrolled into a day-by-day format, but it works reliably. This hashing approach is resource efficient and allows us to create historical feeds of arbitrary duration while minimizing the output size.

Example output calendar_dates.txt:

<table>
<thead>
<tr>
<th>Service ID</th>
<th>Date</th>
<th>Exception Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>BA:2210503:8c4ecb</td>
<td>2020-04-24</td>
<td>1 (Added)</td>
</tr>
<tr>
<td>BA:2210503:8c4ecb</td>
<td>2020-04-23</td>
<td>1 (Added)</td>
</tr>
<tr>
<td>BA:2210503:a4bf1a</td>
<td>2020-04-22</td>
<td>1 (Added)</td>
</tr>
</tbody>
</table>

In this way, the 8c4ecb version of the trip is scheduled to run on the two days of input data where it was seen, and the a4bf1a version is scheduled to run on the other day.

### E.1.4 Differences between Regional and Historic feeds

June 26, 2020
Historic Regional Feeds are equivalent to the original daily Regional Feeds in the stops, routes, and scheduled services they contain. Using a Historic will produce the same output in a routing engine or another type of analysis.

Historic Feeds are different from Regional Feeds in their specific GTFS structure:

- calendars.txt records are removed and rewritten in calendar_dates.txt
- trips.txt records are hashed and compared (as described above)
- IDs for global records are namespaced (as described above)

These differences should not affect routing engine or similar types of analysis. However, keep these differences in mind if you are trying to use historical feeds to understand changes in GTFS data and its practices over time at Bay Area agencies.