BS EN 15531-3:2015

BSI Standards Publication

Public transport — Service interface for real-time information relating to public transport operations

Part 3: Functional service interfaces

bsi.

...making excellence a habit.™

National foreword

This British Standard is the UK implementation of EN 15531-3:2015. It supersedes DD CEN/TS 15531-3:2007 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee EPL/278, Intelligent transport systems.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

ISBN 978 0 580 83399 1

ICS 35.240.60

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 August 2015.

Amendments issued since publication

Date	Text affected

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 15531-3

August 2015

ICS 35.240.60

Supersedes CEN/TS 15531-3:2007

English Version

Public transport - Service interface for real-time information relating to public transport operations - Part 3: Functional service interfaces

Transport public - Interface de service pour les informations en temps réel relatives aux opérations de transport public - Partie 3 : Modules d'interface d'application individuels

Öffentlicher Verkehr - Serviceschnittstelle für Echtzeitinformationen, bezogen auf Operationen im öffentlichen Verkehr - Teil 3: Funktionelle Serviceschnittstelle

This European Standard was approved by CEN on 20 June 2015.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Contents

Page

European foreword

This document (EN 15531-3:2015) has been prepared by Technical Committee CEN/TC 278 “Intelligent transport systems”, the secretariat of which is held by NEN.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by February 2016 and conflicting national standards shall be withdrawn at the latest by February 2016.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document supersedes CEN/TS 15531-3:2007.

This presents Part 3 of the European European Standard known as “SIRI”. SIRI provides a framework for specifying communications and data exchange protocols for organizations wishing to exchange Real-time Information (RTI) relating to public transport operations.

The SIRI European Standard is presented in three parts:

— Context and framework, including background, scope and role, normative references, terms and definitions, symbols and abbreviations, business context and use cases (Part 1).

— The mechanisms to be adopted for data exchange communications links (Part 2).

— Data structures for a series of individual application interface modules (Part 3).

— Two additional parts define additional functional services as CEN Technical Standards:

— Additional data structures for additional application interface module FM (Facility Monitoring: Part 4).

— Additional data structures for additional application interface module SX (Situation eXchange: Part 5).

The XML schema can be downloaded from http://www.siri.org.uk/, along with available guidance on its use, example XML files, and case studies of national and local deployments.

It is recognized that SIRI is not complete as it stands, and from time to time will need to continue to be enhanced to add additional capabilities. It is therefore intended that a SIRI Management Group should continue to exist, at European level, based on the composition of SG7.

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

Introduction

Public transport services rely increasingly on information systems to ensure reliable, efficient operation and widely accessible, accurate passenger information. These systems are used for a range of specific purposes: setting schedules and timetables; managing vehicle fleets; issuing tickets and receipts; providing real-time information on service running, and so on.

This European Standard specifies a Service Interface for Real-time Information (SIRI) about Public Transport. It is intended to be used to exchange information between servers containing real-time public transport vehicle or journey time data. These include the control centres of transport operators and information systems that utilise real-time vehicle information, for example, to deliver services such as travel information. SIRI 2 has also added, with SITI Lite, a possible exchange between server and end-user devices like smartphones or web browsers.

Well-defined, open interfaces have a crucial role in improving the economic and technical viability of Public Transport Information Systems of all kinds. Using standardized interfaces, systems can be implemented as discrete pluggable modules that can be chosen from a wide variety of suppliers in a competitive market, rather than as monolithic proprietary systems from a single supplier. Interfaces also allow the systematic automated testing of each functional module, vital for managing the complexity of increasing large and dynamic systems. Furthermore, individual functional modules can be replaced or evolved, without unexpected breakages of obscurely dependent function.

This European Standard will improve a number of features of public transport information and service management:

— Interoperability – the European Standard will facilitate interoperability between information processing systems of the transport operators by: (i) introducing common architectures for message exchange; (ii) introducing a modular set of compatible information services for real-time vehicle information; (iii) using common data models and schemas for the messages exchanged for each service; and (iv) introducing a consistent approach to data management.

— Improved operations management – the European Standard will assist in better vehicle management by (i) allowing the precise tracking of both local and roaming vehicles; (ii) providing data that can be used to improve performance, such as the measurement of schedule adherence; and (iii) allowing the distribution of schedule updates and other messages in real-time.

— Delivery of real-time information to end-users – the European Standard will assist the economic provision of improved data by; (i) enabling the gathering and exchange of real-time data between VAMS systems; (ii) providing standardized, well defined interfaces that can be used to deliver data to a wide variety of distribution channels.

Technical advantages include the following:

— Reusing a common communication layer for all the various technical services enables cost-effective implementations, and makes the European Standard readily extensible in future.

1 Scope

There are many potential ways for passenger transport operations centres to interact. The approach taken by SIRI is for an open-ended set of standard data structures, carried over a communications channel constructed using one of a small number of specific options.

Part 2 of this European Standard specifies the communications channel. Part 3 specifies a number of functional modules, based on the ‘use cases’ identified in Annex B to Part 1:

— Production Timetable (PT): this service enables the provision of information on the planned progress of vehicles operating a specific service, identified by the vehicle time of arrival and departure at specific stops on a planned route for a particular Operational Day.

— Estimated Timetable (ET): this service enables the provision of information on the actual progress of Vehicle Journeys operating specific service lines, detailing expected arrival and departure times at specific stops on a planned route. There will be recorded data for stops which have been passed, and predicted data for stops not yet passed. In addition the Estimated Timetable service allows Vehicle Journeys to be cancelled, added or changed.

— Stop Timetable (ST): this service provides a stop-centric view of timetabled vehicle arrivals and departures at a designated stop. It can be used to reduce the amount of information that needs to be transmitted in real-time to stops and displays, as reference data for a Stop Monitoring Service; and provides a data feed of the static timetables.

— Stop Monitoring (SM): this service provides a stop-centric view of vehicle arrivals and departures at a designated stop. It can be used by displays and other presentation services to provide departure board and other presentations of timetable and real-time journey information both at stops and at a distance.

— Vehicle Monitoring (VM): this service enables the provision of information on the current location and status of a set of vehicles. It provides all the current relevant information from one AVMS relating to all vehicles fulfilling a set of selection criteria.

— Connection Timetable (CT): this service may be used to provide information about the scheduled arrivals of a feeder vehicle to the operator of a connecting distributor service. The distributor operator can then plan how to guarantee the connection, either with the expected vehicle or a different vehicle.

— Connection Monitoring (CM): this service is used to provide information about the expected arrival of a feeder vehicle to the operator of a connecting distributor service. The distributor operator can then manage the service to guarantee the connection, based on actual vehicle running.

— General Message (GM): the SIRI "General Message” service is used to exchange informative messages between identified individuals in free or an arbitrary structured format. It enables messages to be sent and to be revoked. Messages are assigned validity periods in addition to the actual content.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 15531-1:2015, *Public transport - Service interface for real-time information relating to public transport operations - Part 1: Context and framework*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in EN 15531-1:2015 apply.

4 Symbols and abbreviations

For the purposes of this document, the symbols and abbreviations given in EN 15531-1:2015 apply.

5 Production Timetable Service [PT]

5.1 Purpose

The SIRI Production Timetable Service transmits daily timetables that include any planned updates that are known about at the time of transmission. The service is used typically to communicate between Scheduling systems and AVMS systems, and also between AVMS systems and intelligent clients of the AVMS system. The timetables exchanged should cover all LINEs covered by the AVMS system.

The SIRI Production Timetable Service is also used to transmit the planned interchanges between journeys, including information about the linking of vehicle journey parts through the interchange, such as whether passengers are able to remain seated in the VEHICLE.

The provision of known updates gives a more accurate data set of journeys for the SIRI Estimated Timetable Service to reference, allowing a more efficient real-time exchange of content. However, the provision of a Production Timetable Service is not absolutely essential for the functioning of the Estimated Timetable service.

Because of the enhanced quality of data given by an increased integration with operational and back-office scheduling systems, SIRI implementations that are able to obtain production timetables should always provide and make use of a SIRI Production Timetable Service.

An AVMS system may be aware of additional operational journeys and CALLs such as dead runs, and layovers. As the schedule information system usually only knows about VEHICLE JOURNEYs that are relevant to the passenger, the AVMS should only transmit passenger carrying VEHICLE JOURNEYs to the schedule information system.

5.2 Capability and Permission Matrices

5.2.1 Capability Matrix

Table 1 shows the set of required and optional capabilities defined for the Production Timetable service.

If the service supports Capability Discovery the ***ProductionTimetableCapabilitiesRequest*** / ***Production-TimetableCapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 1 — ProductionTimetableCapabilities Matrix

ProductionTimetableCapabilities				*+Structure*	Capabilities describing implementation of Production Timetable service
inherit	***:::***		**1:1**	*See xxx-Capability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***		0:1	*+Structure*	Which optional filtering features are supported?
		FilterByValidityPeriod	**1:1**	*xsd:boolean*	Whether results can be filtered by Validity Period. Required Capability: Fixed is *true*.
		FilterByOperatorRef	**1:1**	*xsd:boolean*	Whether results can be filtered by OPERATOR. Default is *'true'*.
		FilterByLineRef	**1:1**	*xsd:boolean*	Whether results can be filtered by LINE. Default is *'true'*.

		FilterByVersionRef	**1:1**	*xsd:boolean*	Whether results can be filtered by TIMETABLE Version. Default is *'true'*.
Request Policy	***RequestPolicy***		0:1	*+Structure*	Which features of Request Policy are supported by the service?
		Language	**1:***	*xsd:language*	National languages used by service.
		Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
				choice	Location reference system for coordinates.
	a	***GmlCoordinateFormat***	0:1	*SrsNameType*	Default coordinate format is given by a GML value.
	b	***WgsDecimalDegrees***		*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
Sub-scription Policy	***SubscriptionPolicy***		0:1	*+Structure*	Which features of Subscription Policy are supported by the service?
		HasIncremental-Updates	0:1	*xsd:boolean*	Whether incremental updates can be specified for updates. Default is *'true'*.
Access Control	***AccessControl***		0:1	*+Structure*	Which optional Access Control features are supported by service?
		RequestChecking	1:1	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
		CheckOperatorRef	0:1	*xsd:boolean*	If access control is supported, whether access control by OPERATOR is supported. Default is *'true'*.
		CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
		CheckConnection-LinkRef	0:1	*xsd:boolean*	If access control is supported, whether access control by CONNECTION link is supported. Default is *'true'*.
any	***Extensions***		0:1	*xsd:any**	Placeholder for user extensions.

5.2.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***ProductionTimetableCapabilitiesResponse*** response can include the access permissions for the requestor participant to access data. Table 2 shows this.

Table 2 — Production Timetable Service — Permissions

ProductionTimetablePermission			*+Structure*	Permissions to use implementation of Production Timetable service.
In-herit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant. See Part 2.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant. See Part 2.
	ConnectionLinkPermissions	0:1	*+Structure*	CONNECTION link permissions for participant. See Part 2.

5.3 ProductionTimetableRequest

5.3.1 ProductionTimetableRequest — Element

The ***ProductionTimetableRequest*** states which timetables should be returned – see Table 3 below.

Table 3 — ProductionTimetableRequest — Attributes

ProductionTimetableRequest			*+Structure*	Request for daily production timetables
Attributes	***version***	**1:1**	*VersionString*	Version identifier of *Production Timetable* Service, e.g. 1.0c
Endpoint Properties	***Request-Timestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 for common properties of SIRI Functional Service Requests.
	Message-Identifier	0:1	*Message-Qualifier*	
Line Topic	***ValidityPeriod***	0:1	*ClosedTimestampRange-Structure*	Start and end of timetable validity (time window) of journeys for which schedules are to be returned. Refers to the departure time at the first stop of each VEHICLE JOURNEY. If blank the configured data horizon will be used.
	Start	**1:1**	*xsd:dateTime*	The (inclusive) start time.
	End	**1:1**	*xsd:dateTime*	The (inclusive) end time.
	Timetable-VersionRef	0:1	*xsd:string*	Communicate only differences to the timetable specified by this version of the timetable.
	OperatorRef	0:*	*→Operator-Code*	Filter the results to include only results for the specified operator or operators. Optional SIRI capability: *TopicFiltering / ByOperator.*
	Lines			
	LineDirection			
	LineRef	0:1	*→LineCode*	Filter the results to include only results for the given LINE or LINEs.
	DirectionRef	0:1	*→Direction-Code*	Filter the results to include only journeys for VEHICLEs running in a specific relative DIRECTION. Optional SIRI capability: *TopicFiltering* / *ByDirection*.
Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *NationalLanguage.*
	Include-Translations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
	Incremental-Updates	0:1	*xsd:boolean*	Whether the producer should return the complete set of current data, or only provide updates to the last timetable returned, i.e. additions, modifications and deletions, as indicated by the ***TimetableVersionRef*.** If *false* each subscription response will contain the full information as specified in this request.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

The ***ProductionTimetableRequest*** can be used in both a direct request, and for a subscription. If used for a subscription, additional ***ProductionTimetableSubscriptionPolicy*** parameters can be specified.

The primary *Topic* term on the request is the time window for which timetables are to be returned. Additional topic filtering on LINE and timetable version is also allowed. If filtering is not specified, all LINEs known to the AVMS are transmitted.

If the timetable version is not available an error code is returned ***NoInfoForTopic***. In this situation a subscription is not set up.

5.3.2 ProductionTimetableRequest — Example

The following is an example of a ***ProductionTimetableRequest***.

```
<ServiceRequest>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <RequestorRef>NADER</RequestorRef>
    <ProductionTimetableRequest version="1.0">
        <RequestTimestamp>2001-12-17T09:30:47-05:00</RequestTimestamp>
        <ValidityPeriod>
            <StartTime>14:20:00</StartTime>
            <EndTime>14:20:00</EndTime>
        </ValidityPeriod>
        <TimetableVersionRef>002</TimetableVersionRef>
        <OperatorRef>Smooth</OperatorRef>
<Lines>
<LineDirection>
        <LineRef>123</LineRef>      <DirectionRef>Outbound</DirectionRef>
<LineDirection>
<LineDirection>
    <LineRef>124</LineRef>
<LineDirection>
</Lines>
    </ProductionTimetableRequest>
</ServiceRequest>
```

5.4 ProductionTimetableSubscriptionRequest

5.4.1 ProductionTimetableSubscriptionRequest — Element

The ***ProductionTimetableSubscriptionRequest*** (Table 4) requests the asynchronous delivery of the information described by a ***ProductionTimetableRequest***. The ***ProductionTimetableSubscriptionRequest*** Policy parameters control the processing of the subscription.

Table 4 — ProductionTimetableSubscriptionRequest Parameters

ProductionTimetableSubscriptionRequest			*+Structure*	Request for a subscription to the SIRI Production Timetable Service.
Identity	***SubscriberRef***	0:1	*→ParticipantCode*	See SIRI Part 2 for common ***SubscriptionRequest*** parameters.
	SubscriptionIdentifier	**1:1**	*SubscriptionQualifier*	
Lease	***InitialTerminationTime***	**1:1**	*xsd:dateTime*	
Request	***ProductionTimetable-Request***	**1:1**	*+Structure*	See ***ProductionTimetableRequest***.
	Extensions	0:1	xsd:any*	Placeholder for user extensions.

5.4.2 ProductionTimetableSubscriptionRequest — Example

The following is an example of a ***ProductionTimetableSubscriptionRequest***.

```
    <SubscriptionRequest>
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <RequestorRef>NADER</RequestorRef>
        <ProductionTimetableSubscriptionRequest>
            <SubscriberRef>NADER</SubscriberRef>
            <SubscriptionIdentifier>0000456</SubscriptionIdentifier>
            <InitialTerminationTime>2001-12-17T09:30:47-05:00</InitialTerminationTime>
            <ProductionTimetableRequest version="1.0">
                <RequestTimestamp>2001-12-17T09:30:47-05:00</RequestTimestamp>
                <ValidityPeriod>
                    <StartTime>14:20:00</StartTime>
                    <EndTime>14:20:00</EndTime>
                </ValidityPeriod>
                <TimetableVersionRef>002</TimetableVersionRef>
                <OperatorRef>Smooth</OperatorRef>
<Lines>
<LineDirection>          <LineRef>123</LineRef>
                <DirectionRef>Outbound</DirectionRef>
<LineDirection>
<LineDirection>
    <LineRef>124</LineRef>
<LineDirection>
</Lines>            </ProductionTimetableRequest>
        </ProductionTimetableSubscriptionRequest>
    </SubscriptionRequest>
```

5.5 ProductionTimetableDelivery

5.5.1 Introduction

The ***ProductionTimetableDelivery*** returns the position of a VEHICLE or group of VEHICLEs.

5.5.2 ServiceDelivery with a ProductionTimetableDelivery

One or more ***ProductionTimetableDelivery*** elements may be returned as part of a SIRI Functional ServiceDelivery. Table 5 shows this element.

Table 5 — ServiceDelivery / ProductionTimetableDelivery — Attributes

ServiceDelivery			*+Structure*	See SIRI Part 2 ***ServiceDelivery***
HEADER	***:::***	**1:1**	See ServiceDelivery	
Payload	***ProductionTimetableDelivery***	0:*	*+Structure*	See ***ProductionTimetableDelivery*** element.

5.5.3 ProductionTimetableDelivery — Element

Each ***ProductionTimetableDelivery*** is made up of ***DatedTimetableVersionFrame*** elements. There will be status messages for any request that could not be returned. Table 6 shows this element.

Table 6 — ProductionTimetableDelivery — Attributes

ProductionTimetableDelivery			*+Structure*	Describes one or more Dated Timetables.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Production Timetable Service. Fixed, e.g. 1.0.
LEADER	***:::***	**1:1**	xxx***ServiceDelivery***	See SIRI Part 2.xxx***ServiceDelivery***.
Payload	***DatedTimetable VersionFrame***	0:*	*+Structure*	A version of the timetable to run on a specified date. See ***DatedTimetableVersionFrame*** element.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

5.5.4 DatedTimetableVersionFrame — Element

5.5.4.1 Introduction

Each production timetable is returned as a ***DatedTimetableVersionFrame*** element. Each ***DatedTimetableVersionFrame*** contains a version of the timetable for a LINE and DIRECTION, and comprises one or more ***DatedVehicleJourney*** elements. Table 7 shows this element.

Table 7 — DatedTimetableVersionFrame — Attributes

DatedTimetableVersionFrame			*+Structure*	Provides a schedule of DATED VEHICLE JOURNEY for a LINE and DIRECTION.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded.
Identity	***VersionRef***	0:1	*→VersionCode*	Reference to a TIMETABLE VERSION FRAME.
Line	***LineRef***	**1:1**	*→LineCode*	Identifier for the LINE.
	DirectionRef	**1:1**	*→DirectionCode*	Reference to the DIRECTION the VEHICLE is running along the LINE, for example, "in" or "out", “clockwise”. Distinct from a destination.
Journey Pattern Info	***:::***	0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfo-Group***.
Service Info	***:::***	0:1	*ServiceInfoGroup*	See SIRI Part 2 ***ServiceInfoGroup***.
Notes	***OriginDisplay***	0:*	*NLString*	Name of ORIGIN of journey. One per language (+SIRI v2.0).
	DestinationDisplay	0:*	*NLString*	Name of DESTINATION of journey. One per language (Unbounded 0:* since +SIRI v2.0).
	LineNote	0:*	*NLString*	Text associated with LINE. One per language (Unbounded 0:* since +SIRI v2.0).
	FirstOrLastJourney	0:1	*unspecified*	Whether journey is first or last jouurney of day. +SIRI v2.0
Real time defaults	***HeadwayService***	0:1	*xsd:boolean*	Whether this is a Headway Service, that is one shown as operating at a prescribed interval rather than to a fixed timetable.
	Monitored	0:1	*xsd:boolean*	Whether there is real-time information available for journey, if not present, not known.
Journeys	***DatedVehicleJourney***	0:*	*+Structure*	Provides schedule information about the VEHICLE JOURNEY along which a VEHICLE is running.
Inter-changes	***ServiceJourney-Interchange***	0:*	*+Structure*	Provides schedule information about the planned SERVICE JOURNEY INTERCHANGEs that connect services.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

5.5.4.2 DatedVehicleJourney — Element

Each ***DatedVehicleJourney*** contains an ordered list of ***DatedCall*** elements representing the sequence of stop CALLs, as well as other properties that apply to the VEHICLE JOURNEY as a whole. Table 8 shows this element.

Table 8 — DatedVehicleJourney — Attributes

DatedVehicleJourney				*+Structure*	Provides schedule information about the VEHICLE JOURNEY along which a VEHICLE is running.
Vehicle Journey Identity	***Choice***	**Framed-Vehicle-JourneyRef**	1:1	*+Structure*	Identifies the DATED VEHICLE JOURNEY. The preferred construction is using FramedVehicle-JourneyRef from SIRI 2.0. For backward compatibility it is still possible to use the deprecated DatedVehicle-JourneyCode
		Dated-Vehicle-Journey-Code	1:1	*VehicleJourney-Code*	
	VehicleJourneyRef		0:1	*→Vehicle-JourneyCode*	VEHICLE JOURNEY from which this journey is different.
	ExtraJourney		0:1	*xsd:boolean*	Whether this journey is an addition to the plan Can only be used when both participants recognise the same schedule version. If omitted, defaults to *false*: the journey is not an addition.
	Cancellation		0:1	*xsd:boolean*	Whether this journey is a Cancellation of a journey in the plan. Can only be used when both participants recognise the same schedule version. If omitted, defaults to *false*: Journey is not a Cancellation.
Journey Pattern Info	***:::***		0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup.***
Service Info	***:::***		0:1	*ServiceInfo-Group*	See SIRI Part 2 ***ServiceInfoGroup.***
Journey-Info Group	***:::***		0:1	*JourneyInfo-Group* See above	See SIRI Part 2 ***JourneyInfoGroup.***
Notes	***OriginDisplay***		0:*	*NLString*	The appropriate text to be used as Origin for this VEHICLE JOURNEY. Can be overridden at individual calling points. (+SIRI v2.0). One per language
	DestinationDisplay		0:*	*NLString*	The appropriate text to be used as Destination for this VEHICLE JOURNEY. Can be overridden at individual calling points. One per language (Unbounded 0:* since +SIRI v2.0).
	LineNote		0:*	*NLString*	Additional text associated with LINE. Inherited property. One per language (Unbounded 0:* since +SIRI v2.0).
	FirstOrLastService		0:1	*xsd:boolean*	Whether journey is first or last jouurney of day. +SIRI v2.0
Timetable-info	***HeadwayService***		0:1	*xsd:boolean*	Whether this is a HEADWAY JOURNEY, that is one shown as operating at a prescribed interval rather than to a fixed timetable.
Real-time Info	***Monitored***		0:1	*xsd:boolean*	Whether there is real-time information available for journey, if not present, not known.
Operation-al Block	***:::***		0:1	*OperationalBlock Group*	See SIRI Part 2 ***OperationalBlockGroup.***
Children	***a***	***DatedCalls***	**1:1**	*+Structure*	Complete sequence of stops along the route path, in calling order. CALLs are in order within a JOURNEY PATTERN.
		DatedCall	**2:***	*+Structure*	Individual ***DatedCall.*** See below.
any	***Extensions***		0:1	*xsd:any**	Placeholder for user extensions.

5.5.4.3 DatedCall — Element

Each ***DatedCall*** describes the PASSING TIMEs of a VEHICLE JOURNEY at a stop, together with other data elements relating to the CALL. A journey shall contain at least two CALLs. Table 9 shows this element.

Table 9 — DatedCall — Attributes

DatedCall			*+Structure*	Provides information about a CALL in a DATED VEHICLE JOURNEY.
Stop Identity	***StopPointRef***	**1:1**	*→StopPointCode*	Reference to a SCHEDULED STOP POINT. Defaults to that of context i.e. that specified on ***StopPoint***.
	VisitNumber	0:1	*VisitNumberType*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to a stop, the ***VisitNumber*** count is used to distinguish each separate visit. Default is *'1'*
	Order	0:1	*xsd:positive-Integer*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Name of Stop. One per language (Unbounded 0:* since +SIRI v2.0).
Info	***TimingPoint***	0:1	*xsd:boolean*	Whether the stop is a TIMING POINT. Times for stops that are not timing points are sometimes interpolated crudely from the timing points, and may represent a lower level of accuracy. Default is *'true'*.
	BoardingStretch	0:1	*xsd:boolean*	Whether this is a Hail and Ride stop. A hail and ride stop may represent a linear stretch in the stop model. Default is *'false'*.
	RequestStop	0:1	*xsd:boolean*	Whether VEHICLE stops only if requested explicitly by passenger. Default is *'false'*.
Service Info	***OriginDisplay***	0:*	*NLString*	The publicised ORIGIN at this point in the journey; used to help identify the VEHICLE to the public. Since VEHICLEs can change their published ORIGIN during a journey, the Origin included here should be what the VEHICLE will display when it reaches this stop. The ORIGIN DISPLAY is specified by exception: if none is specified, the value from any previous ***DatedCall*** instances for the same journey will be used. If there are none then any values from the underlying ***DatedVehicleJourney*** will be used, or failing that, from the ***DatedTimetableVersionFrame***. One per language (Unbounded 0:* since +SIRI v2.0).
	DestinationDisplay	0:*	*NLString*	The publicised DESTINATION at this point in the journey; used to help identify the VEHICLE to the public. Since VEHICLEs can change their published DESTINATION during a journey, the destination included here should be what the VEHICLE will display when it reaches this stop. The DESTINATION DISPLAY is specified by exception: if none is specified, the value from any previous ***DatedCall*** instances for the same journey will be used. If there are none then any values from the underlying ***DatedVehicleJourney*** will be used, or failing that, from the ***DatedTimetableVersionFrame***. One per language (Unbounded 0:* since +SIRI v2.0).

Call	***CallNote***	0:*	*NLString*	Text annotation that applies to this CALL. One per language (Unbounded 0:* since +SIRI v2.0).
Disruption-Group	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***DisruptionGroup***. *DetailLevel*: normal.
Arrival	***AimedArrivalTime***	0:1	*xsd:dateTime*	Aimed Arrival Time in either the original or Production Timetable. Can be omitted at the end stop.
	Arrival-PlatformName	0:1	*NLString*	Bay or platform name. Inherited property. Can be omitted if the same as the ***DeparturePlatformName***.
	Arrival-BoardingActivity	0:1	*ArrivalBoarding-ActivityEnum*	Type of alighting allowed at stop. Default is '*alighting'.* *alighting* | *noAlighting* | *passthru*
	ArrivalStop-Assignment	0:1	*+Structure*	Assignment of arrival of Scheduled STOP POINT to a physical QUAY (platform). If not given, assume same as for departure +SIRI v2.0.
	AimedQuayRef	0:1	*→QuayCodeType*	Physical QUAY to use according to the planned timetable. +SIRI v2.0
	Aimed-QuayName	0:1	*NLString*	Scheduled Platform name. Can be used to indicate a platform change. +SIRI v2.0.
Departure	***Aimed-DepartureTime***	0:1	*xsd:dateTime*	Aimed Departure Time in either the original or Production Timetable.
	Departure-PlatformName	0:1	*NLString*	Bay or platform (QUAY) name from which VEHICLE will depart.
	Departure-BoardingActivity	0:1	*Departure-BoardingActivityEnum*	Type of boarding allowed at stop. Default is '*boarding'.* *boarding* | *noBoarding*| *passthru*
	DepartureStop-Assignment	0:1	*+Structure*	Assignments of departure platform for SCHEDULED STOP POINT to a physical QUAY. +SIRI v2.0. +SIRI v2.0. *DetailLevel*: normal.
	AimedQuayRef	0:1	*→QuayCodeType*	Physical QUAY (Platform) to use according to the planned timetable. +SIRI v2.0
	Aimed-QuayName	0:*	*NLString*	Scheduled QUAY (Platform) name. Can be used to indicate a platform change. +SIRI v2.0 One per language
	AimedLatestPassengerAccessTime	0:1	*xsd:dateTime*	Latest target time at which a PASSENGER should aim to arrive at the STOP PLACE containing the stop. This time may be earlier than the VEHICLE departure times and may include time for processes such as checkin, security, etc.(As specified by CHECK CONSTRAINT DELAYs in the underlying data) If absent assume to be the same as Earliest expected departure time, +SIRI 2.0
Headway	***AimedHeadway-Interval***	0:1	*Positive-DurationType*	Target headway interval between services for frequency based services.
Interchange	***Targeted-Interchange***	0:*	*+Structure*	Information on any planned connections. If omitted: No connections, i.e. ***TargetedInterchange***. *Deprecated from SIRI v2.0*
Interchange	***FromServiceJourneyInterchange***	0:*	*+Structure*	Information on any feeder of a planned connections. +SIRI v2.0
Interchange	***ToServiceJourneyInterchange***	0:*	*+Structure*	Information on any distributor of a planned connections. +SIRI v2.0
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

5.5.4.4 TargetedInterchange — Element

Each ***TargetedInterchange*** describes the SERVICE JOURNEY INTERCHANGEs (i.e. connections) that may be made at a stop to another onwards distributor journey. Each ***TargetedInterchange*** is described in the context of a specific ***DatedCall*** for a specific journey. Table 10 shows this element.

Table 10 — TargetedInterchange — Attributes

TargetedInterchange			*+Structure*	Information on any planned connections. If omitted: No connections.
Identity	***InterchangeCode***	0:1	*→Interchange-Code*	Identifier of SERVICE JOURNEY INTERCHANGE.
	DistributorVehicle-JourneyRef	**1:1**	*→DatedVehicle-JourneyCode*	Reference to a distributor DATED VEHICLE JOURNEY.
Connection	***Distributor-ConnectionLink***	**1:1**	*+Structure*	CONNECTION link over which SERVICE JOURNEY INTERCHANGE takes place.
	ConnectionCode	**1:1**	*ConnectionCode*	Identifier of CONNECTION link.
	StopPointRef	0:1	*→StopPoint-Code*	SCHEDULED STOP POINT from which the distributor VEHICLE JOURNEY departs. If omitted: the distributor journey stop is the same as the feeder journey stop, i.e. that of the containing dated CALL.
	Interchange-Duration	0:1	*PositiveDuration-Type*	Time (Duration) needed by passenger to change from feeder to distributor.
	Frequent-TravellerDuration	0:1	*PositiveDuration-Type*	Time (Duration) needed by passenger who is familiar with CONNECTION link to change from feeder to distributor. If absent, use ***InterchangeDuration*** and a standard weighting.
	Occasional-TravellerDuration	0:1	*PositiveDuration-Type*	Time (Duration) needed by passenger who is not familiar with CONNECTION link to change from feeder to distributor. If absent, use ***InterchangeDuration*** and a standard weighting.
	ImpairedAccess-Duration	0:1	*PositiveDuration-Type*	Time (Duration) needed by impaired mobility passenger to change from feeder to distributor. If absent, use ***InterchangeDuration*** and a standard weighting.
Identity	***Distributor-VisitNumber***	0:1	*VisitNumber-Type*	Sequence of visit to stop within distributor VEHICLE JOURNEY. Increases monotonically, but not necessarily sequentially.
	DistributorOrder	0:1	*xsd:positive-Integer*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
Interchange Properties	***StaySeated***	0:1	*xsd:boolean*	Whether Vehicle stops only if requested explicitly by passenger. Default is *'false'*.
	Guaranteed	0:1	*xsd:boolean*	Whether the SERVICE JOURNEY INTERCHANGE is guaranteed. Default is *false*; interchange is not guaranteed.
	Advertised	0:1	*xsd:boolean*	Whether the SERVICE JOURNEY INTERCHANGE is advertised as a guaranteed connection. Default is *'false'*.

Interchange Times	***StandardWaitTime***	0:1	*xsd:duration*	Standard wait time for INTERCHANGE. SIRI v2.0
	MaximumWaitTime	0:1	*xsd:duration*	Maximum time that Distributor will wait for Feeder for INTERCHANGE. SIRI v1.0.
	MaximumAutomatic-WaitTime	0:1	*xsd:duration*	Maximum automatic wait time that Distributor will wait for Feeder for INTERCHANGE. SIRI v2.0.
	StandardTransfer-Time	0:1	*xsd:duration*	Standard transfer duration for INTERCHANGE. SIRI
	MinimumTransfer-Time	0:1	*xsd:duration*	Minimum transfer duration for INTERCHANGE. SIRI
	MaximumTransfer-Time	0:1	*xsd:duration*	Maximum transfer duration for INTERCHANGE. SIRI
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

5.5.4.5 ServiceJourneyInterchange — Element (SIRI v2.0)

Each ***ServiceInterchange*** describes the SERVICE JOURNEY INTERCHANGEs, i.e. connections that may be made over a CONNECTION link between a feeder and a distributor journey. The element is similar in purpose to ***TargetedInterchange*** but also allows feeder connections to be described, as well as the exchange of SERVICE JOURNEY INTERCHANGEs independent of the exchange of a specific journey. Table 11 shows this element. Note that redundant Feeder and Distributor information may be excluded when given by context.

Table 11 — ServiceJourneyInterchange — Attributes

ServiceJourneyInterchange				*+Structure*	Information on any planned connections, i.e. SERVICE JOURNEY INTERCHANGEs. If omitted: No connections.
Identity	***InterchangeCode***		0:1	*→Interchange-Code*	Identifier of SERVICE JOURNEY INTERCHANGE.
Connection Link Identity	***ConnectionLinkRef***		0:1		Reference to a physical CONNECTION LINK over which the SERVICE JOURNEY INTERCHANGE takes place. +SIRI v2.0
Feeder Vehicle Journey Identity	***FeederRef***		0:1*	*+Structure*	Reference to a connecting feeder VEHICLE JOURNEY. +SIRI v2.0. *)This structure is mandatory except when the ServiceJourneyInterchange is in context of the feeder's DATED CALL.
		FramedVehicle-JourneyRef	0:1	*+Structure*	Frame qualified Reference to a connecting feeder VEHICLE JOURNEY. +SIRI v2.0
		DataFrameRef	0:1	*→Vehicle-JourneyCode*	Reference to a data frame within participant service. Used to ensure that the ***DatedVehicleJourneyRef*** is unique with the data horizon of the service. Often the ***OperationalDayType*** is used for this purpose. +SIRI v2.0
		DatedVehicle-JourneyRef	0:1	*→DatedVehicle-JourneyCode*	A reference to the DATED VEHICLE JOURNEY that the VEHICLE is making. +SIRI v2.0
		LineRef	0:1	*→LineCode*	A reference to the LINE of the referenced feeder VEHICLE JOURNEY. +SIRI v2.0
		ParticipantRef	0:1	*→Participant-Code*	A reference to the system from which the feeder reference comes.
Feeder StopPoint Identity	***FeederArrivalStopRef***		0:1*	*→StopPointRef*	SCHEDULED STOP POINT at which feeder journey arrives. +SIRI v2.0. *)This element is mandatory except when the ServiceJourneyInterchange is in context of the distributors dated Call.

	FeederVisitNumber	0:1	*VisitNumber-Type*	Sequence of visit to stop within feeder VEHICLE JOURNEY. Increases monotonically, but not necessarily sequentially.
Distributor Vehicle Journey Identity	***DistributorRef***	0:1*	*+Structure*	Reference to a connecting distributor VEHICLE JOURNEY. +SIRI v2.0. *)This structure is mandatory except when the ServiceJourneyInterchange is in context of the distributor's DATED CALL.
	FramedVehicle-JourneyRef	0:1	*+Structure*	Frame qualified Reference to a connecting distributor VEHICLE JOURNEY. +SIRI v2.0
	DataFrameRef	0:1	*→Vehicle-JourneyCode*	Reference to a data frame within participant service. Used to ensure that the ***DatedVehicleJourneyRef*** is unique with the data horizon of the service. Often the ***OperationalDayType*** is used for this purpose. +SIRI v2.0
	DatedVehicle-JourneyRef	0:1	*→DatedVehicle-JourneyCode*	A reference to the distributor DATED VEHICLE JOURNEY that the VEHICLE is making. +SIRI v2.0
	LineRef	0:1	*→LineCode*	A reference to the LINE of the referenced distributor VEHICLE JOURNEY. +SIRI v2.0
	ParticipantRef	0:1	*→Participant-Code*	A reference to the system from which the distributor reference comes
Distributor Stop Point Identity	***DistributorDepartureStopRef***	0:1*	*→StopPointRef*	SCHEDULED STOP POINT at which distributor journey departs. +SIRI v2.0. *)This element is mandatory except when the ServiceJourneyInterchange is in context of the distributor's DATED CALL..
	DistributorVisitNumber	0:1	*VisitNumber-Type*	Sequence of visit to stop within distributor VEHICLE JOURNEY. Increases monotonically, but not necessarily sequentially.
Interchange Properties	***StaySeated***	0:1	*xsd:boolean*	Whether Vehicle stops only if requested explicitly by passenger. Default is *'false'*.
	Guaranteed	0:1	*xsd:boolean*	Whether the interchange is guaranteed. Default is *false*; interchange is not guaranteed.
	Advertised	0:1	*xsd:boolean*	Whether the interchange is advertised as a connection. Default is *'false'*.
Interchange Times	***StandardWaitTime***	0:1	*xsd:duration*	Standard wait time for INTERCHANGE. SIRI v2.0
	MaximumWaitTime	0:1	*xsd:duration*	Maximum time that Distributor will wait for Feeder for INTERCHANGE. SIRI v1.0.
	MaximumAutomatic-WaitTime	0:1	*xsd:duration*	Maximum automatic wait time that Distributor will wait for Feeder for INTERCHANGE. SIRI v2.0.
	StandardTransferTime	0:1	*xsd:duration*	Standard transfer duration for INTERCHANGE. SIRI v2.0
	MinimumTransferTime	0:1	*xsd:duration*	Minimum transfer duration for INTERCHANGE. SIRI v2.0
	MaximumTransferTime	0:1	*xsd:duration*	Maximum transfer duration for INTERCHANGE. SIRI v2.0
Any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

5.5.5 ProductionTimetableDelivery — Example

The following is an example of a ***ProductionTimetableDelivery***.

```
<ServiceDelivery>
    <!--=======HEADER================================================= -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
```

```
<Status>true</Status>
<MoreData>false</MoreData>
<!--=======PAYLOAD================================================ -->
<ProductionTimetableDelivery version="1.0">
    <ResponseTimestamp>2001-12-17T09:30:47-05:00</ResponseTimestamp>
    <SubscriberRef>NADER</SubscriberRef>
    <SubscriptionRef>00123456</SubscriptionRef>
    <Status>true</Status>
    <ValidUntil>2001-12-17T10:30:47-05:00</ValidUntil>
    <DatedTimetableVersionFrame>
        <LineRef>123</LineRef>
        <DirectionRef>Out</DirectionRef>
        <PublishedLineName>String</PublishedLineName>
        <DirectionName xml:lang="EN">Outbound</DirectionName>
        <!-- Service Info Group -->
        <OperatorRef>Smooth</OperatorRef>
        <ProductCategoryRef>Fun Special</ProductCategoryRef>
        <ServiceFeatureRef>normalStopping</ServiceFeatureRef>
        <VehicleFeatureRef>lowStep</VehicleFeatureRef>
        <ServiceFeatureRef>cyclesPermitted</ServiceFeatureRef>
        <LineNote>Special services at Easter</LineNote>
        <!-- Timetable real-time info Group -->
        <Monitored>true</Monitored>
        <DatedVehicleJourney>
            <!-- Identity Group -->
            <DatedVehicleJourneyCode>DVC0008767</DatedVehicleJourneyCode>
            <VehicleJourneyRef>VJ123</VehicleJourneyRef>
            <ExtraJourney>false</ExtraJourney>
            <PublishedLineName>123 Out</PublishedLineName>
            <!-- Service Info Group -->
            <OperatorRef>Sharp</OperatorRef>
            <ProductCategoryRef>plusbus</ProductCategoryRef>
            <ServiceFeatureRef>hailAndRider</ServiceFeatureRef>
            <VehicleFeatureRef>lowDoor</VehicleFeatureRef>
            <!-- Journey Info Group -->
            <VehicleJourneyName>Shoppers Special</VehicleJourneyName>
            <JourneyNote>Not suitable for claustrophobes.</JourneyNote>
            <Monitored>true</Monitored>
            <HeadwayService>false</HeadwayService>
            <!-- == CALL 1 == -->
            <DatedCall>
                <StopPointRef>HLTS00101</StopPointRef>
                <BoardingStretch>false</BoardingStretch>
                <RequestStop>false</RequestStop>
                <DestinationDisplay>Limbo</DestinationDisplay>
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
                <ArrivalBoardingActivity>alighting</ArrivalBoardingActivity >
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
                <DeparturePlatformName>1</DeparturePlatformName>
                <DepartureBoardingActivity>boarding</ DepartureBoardingActivity >
            </DatedCall>
            <!-- == CALL 2 ==-->
            <DatedCall>
                <StopPointRef>HLTS00102</StopPointRef>
                <BoardingStretch>true</BoardingStretch>
                <DestinationDisplay>Hell First Circle</DestinationDisplay>
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
                <AimedDepartureTime>2001-12-17T09:30:47-05:00</AimedDepartureTime>
                <TargetedInterchange>
                    <DistributorVehicleJourneyRef>V45681</DistributorVehicleJourneyRef>
                    <DistributorConnectionLink>
                        <ConnectionLinkCode>01340</ConnectionLinkCode>
                    <DefaultDuration>P3M</DefaultDuration>
                    </DistributorConnectionLink>
                    <DistributorVisitNumber>1</DistributorVisitNumber>
                    <Guaranteed>true</Guaranteed>
                </TargetedInterchange>
            </DatedCall>
```

```
                    <!-- == CALL 3 == -->
                    <DatedCall>
                        <StopPointRef>HLTS00103</StopPointRef>
                        <DestinationDisplay>Hell</DestinationDisplay>
                        <CallNote>Can change here to </CallNote>
                        <TimetabledArrivalTime>2001-12-17T09:30:47-05:00</TimetabledArrivalTime>
                        <ArrivalPlatformName>4</ArrivalPlatformName>
                        <TimetabledDepartureTime>2001-12-17T09:30:47-05:00</TimetabledDepartureTime>
                        <DepartureBoardingActivity>NoBoarding</DepartureBoardingActivity>
                        <TargetedInterchange>
                            <InterchangeCode>I765</InterchangeCode>
                            <DistributorVehicleJourneyRef>V45678</DistributorVehicleJourneyRef>
                            <DistributorConnectionLink>
                                <ConnectionLinkCode>01345</ConnectionLinkCode>
                                <StopPointRef>HLTS00103</StopPointRef>
                                <StopPointName>Gare de Nord</StopPointName>
                                <DefaultDuration>P5M</DefaultDuration>
                                <FrequentTravellerDuration>P3M</FrequentTravellerDuration>
                                <OccasionalTravellerDuration>P7M</OccasionalTravellerDuration>
                                <ImpairedAccessDuration>P15M</ImpairedAccessDuration>
                            </DistributorConnectionLink>
                            <DistributorVisitNumber>2</DistributorVisitNumber>
                            <StaySeated>false</StaySeated>
                            <Guaranteed>true</Guaranteed>
                            <Advertised>true</Advertised>
                            <MaximumWaitTime>P15M</MaximumWaitTime>
                        </TargetedInterchange>
                    </DatedCall>
                </DatedVehicleJourney>
            </DatedTimetableVersionFrame>
        </ProductionTimetableDelivery>
    </ServiceDelivery>
```

5.5.6 Efficient Delivery of Large Timetables

Production Timetable Service responses may be very large. The XML messages may be broken down into smaller packets for transmission using the ***MoreData*** attribute in the ***DataSupplyAnswer*** message. See Multipart Despatch earlier.

In order to reduce the data volume it is possible to exchange only the changes instead of the entire schedule. However, for this to work the daily updated schedule of the AVMS has to originate from the same database as the seasonal schedule of the schedule information system. Note also that the daily updated planned schedules usually exist as a progression of the seasonal schedule, which means that over time, the deviations from the originally created seasonal schedule become greater. To obtain just changes, the active schedule version against which changes are desired shall be specified in the ***TimetableVersionRef*** of the subscription request. If the same or an earlier version is active in the data sender, then it need only transmit the changes to the planned schedule in the Production Timetable service.

6 Estimated Timetable Service [ET]

6.1 Purpose

The Estimated Timetable service is used by the AVMS to inform interested schedule information systems of the current status of all known VEHICLE JOURNEYs. This enables the schedule information system to provide up-to-the-minute information for short-term journey planning. It can also be used to support intelligent displays that calculate the deviation from timetable themselves using a timetable and a real time difference delay by the SIRI Stop Monitoring Service.

Message content can include:

— Schedule deviations with predictions for future route sections.

— Failure of a VEHICLE JOURNEY.

— Additional journeys.

— Change to the VEHICLE capacity (passenger load).

— Change to VEHICLE TYPE/VEHICLE Equipment.

— Platform changes.

— Stop obstruction.

— Path (i.e. JOURNEY PATTERN) changes.

As with the production timetable service, the AVMS shall only transmit the boardable VEHICLE JOURNEYs, i.e. the VEHICLE JOURNEYs that carry passengers.

6.2 Capability and Permission Matrices

6.2.1 Capability Matrix

The following set of required and optional capabilities is defined for the Estimated Timetable service. If the service supports Capability Discovery, the ***EstimatedTimetableCapabilitiesRequest*** / ***EstimatedTimetable-CapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 12 — EstimatedTimetable Service Capabilities Matrix

EstimatedTimetableCapabilities				*+Structure*	Capabilities describing implementation of Estimated Timetable service.
inherit	*:::*		**1:1**	*See xxx-Capability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***		0:1	*+Structure*	Which optional filtering features are supported?
		DefaultPreview-Interval	0:1	*Positive-DurationType*	Default Preview Interval.
		FilterByOperator-Ref	0:1	*xsd:boolean*	Whether results can be filtered by OPERATOR. Required Capability: default is *'true'*.
		FilterByLineRef	0:1	*xsd:boolean*	Whether results can be filtered by LINE. Required Capability: Default is *'true'*.
		FilterByVersionRef	0:1	*xsd:boolean*	Whether results can be filtered by TIMETABLE Version. Default is *'true'*.
Request Policy	***RequestPolicy***		0:1	*+Structure*	Which optional features of the Request Policy are supported by the service?
		Language	**1:***	*xsd:language*	National languages used by service.
				choice	Location reference coordinate system.
		Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
	a	***GmlCoordinate-Format***	0:1	*SrsNameType*	Default coordinate format is given by a GML format name value.
	b	***WgsDecimal-Degrees***		*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude in decimal degrees of arc.
Sub-scription	***SubscriptionPolicy***		0:1	*+Structure*	Which features of the ***SubscriptionPolicy*** are supported by service?

Policy		***HasIncremental-Updates***	0:1	*xsd:boolean*	Whether incremental updates can be specified for updates. Default is *'true'*.
		HasChange-Sensitivity	0:1	*xsd:boolean*	Whether change threshold can be specified for updates. Default is *'true'*.
Access Control	***AccessControl***		0:1	*+Structure*	Which optional Access Control features are supported by service?
		RequestChecking	0:1	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
		CheckOperatorRef	0:1	*xsd:boolean*	If access control is supported, whether access control by OPERATOR is supported. Default is *'true'*.
		CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
		CheckConnection-LinkRef	0:1	*xsd:boolean*	If access control is supported, whether access control by CONNECTION link is supported. Default is *'true'*.
any	***Extensions***		0:1	xsd:any*	Placeholder for user extensions.

6.2.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***EstimatedTimetable-CapabilitiesResponse*** response can include the access permissions for the requestor participant to access data.

Table 13 — EstimatedTimetable Service — Permissions

EstimatedTimetablePermission			*+Structure*	Permissions to use implementation of Estimated Timetable service.
Inherit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant. See Part 2.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant. See Part 2.
	ConnectionLinkPermissions	0:1	*+Structure*	CONNECTION link permissions for participant. See Part 2.

6.3 EstimatedTimetableRequest

6.3.1 EstimatedTimetableRequest — Element

The ***EstimatedTimetableRequest*** states which timetables should be returned.

The ***EstimatedTimetableRequest*** can be used in both a direct request, and for a subscription. If used for a subscription, additional ***EstimatedTimetableSubscriptionPolicy*** parameters can be specified.

The primary *Topic* term on the request is the time window for which timetables are to be returned. Additional topic filtering on LINE is also allowed. If filtering is not specified, all LINEs known to the AVMS are transmitted.

If the timetable version is not available, an error code is returned NoInfoForTopic. In this situation a subscription is not set up.

Table 14 — EstimatedTimetableRequest — Attributes

EstimatedTimetableRequest			*+Structure*	Request for Estimated timetables.
Attributes	***Version***	**1:1**	*VersionString*	Version identifier of Estimated Timetable Service, e.g. '1.0c.'
Endpoint Properties	***RequestTimestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	MessageIdentifier	0:1	*MessageQualifier*	
Topic	***PreviewInterval***	0:1	*PositiveDurationType*	Maximum Preview time in minutes requested by the schedule information system. Preview horizon only limits real time prediction data, information on additional journeys, cancellations, etc., that are beyond the interval may still be included.
	TimetableVersionRef	0:1	*xsd:string*	Communicate only differences to the timetable specified by this version of the timetable.
	OperatorRef	0:1	*→OperatorCode*	Filter the results to include only results for the specified operator. Optional SIRI capability: *TopicFiltering / ByOperator.*
	Lines	0:*	*LineDirection*	Filter the results to include only VEHICLEs along the given LINEs. SIRI v1.3
	LineRef	0:1	*→LineCode*	Filter the results to include only results for the given LINE. If no ***LineRef*** is specified as a subscription filter, this implicitly implies transmission of data for all LINEs in the AVMS.
	DirectionRef	0:1	*→DirectionCode*	Filter the results to include only Stop Visits for VEHICLEs running in a specific relative DIRECTION, for example, "inbound" or "outbound". (Direction does not specify a destination.) Optional SIRI capability: *TopicFiltering / ByDirection.*
Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *NationalLanguage*.
	IncludeTranslations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
	EstimatedTimetableDetailLevel	0:1	*EstimatedTimetableDetailLevelEnum*	Level of detail to include in response. *minimum* | *basic* | *normal* | *calls* | *full* Default is '*calls*. Optional SIRI capability: *DetailLevel* (if absent, must support *normal)*. +SIRI.2.0
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

6.3.2 EstimatedTimetableRequest — Example

The following is an example of an ***EstimatedTimetableRequest***.

```
<ServiceRequest>
        <EstimatedTimetableRequest version="1.0">
            <RequestTimestamp>2001-12-17T09:30:47-05:00</RequestTimestamp>
            <PreviewInterval>P1Y2M3DT10H30M</PreviewInterval>
            <TimetableVersionRef>0008</TimetableVersionRef>
            <OperatorRef>SMOOTH</OperatorRef>
```

```
<Lines
<LineDirection>
                    <LineRef>123</LineRef>
                    <DirectionRef>INBOUND</DirectionRef>
</LineDirection>
</Lines>
              </EstimatedTimetableRequest>
     </ServiceRequest>
```

6.4 EstimatedTimetableSubscriptionRequest

6.4.1 EstimatedTimetableSubscriptionRequest — Element

The ***EstimatedTimetableSubscriptionRequest*** requests the asynchronous delivery of the information described by an ***EstimatedTimetableRequest***. The ***EstimatedTimetableSubscriptionRequest*** Policy parameters control the processing of the subscription.

At the start of the subscription, the AVMS is responsible for transmitting the current full status to the schedule information system, i.e. it transmits all real-time data that deviates from the planned schedule at the start of the subscription.

Table 15 — EstimatedTimetableSubscriptionRequest — Attributes

EstimatedTimetable-SubscriptionRequest			*+Structure*	Request for a subscription to the SIRI Estimated Timetable Service.
Identity	***SubscriberRef***	0:1	*→ParticipantCode*	See SIRI Part 2, Common ***SubscriptionRequest*** parameters.
	SubscriptionIdentifier	**1:1**	*Subscription-Qualifier*	
Lease	***InitialTerminationTime***	**1:1**	*xsd:dateTime*	
Request	***EstimatedTimetable-Request***	**1:1**	*+Structure*	See ***EstimatedTimetableRequest***.
Policy	***IncrementalUpdates***	**0:1**	*boolean*	Whether the producer should return the complete set of data, or only provide updates to the previously returned data i.e. changes to the expected deviation (delay or early time). Default is 'true'. If true only changes at the first stop will be returned and the client must interpolate the If false each subscription response will contain the full information as specified in this request. (SIRI 2.0)
	ChangeBeforeUpdate	**0:1**	*PositiveDuration-Type*	If incremental update, Threshold value in seconds after which the deviations from planned schedule or the last message are to be transmitted. (Optional from SIRI 2.0)
	Extensions	0:1	xsd:any*	Placeholder for user extensions.

6.4.2 EstimatedTimetableSubscriptionRequest — Example

The following is an example of an ***EstimatedTimetableSubscriptionRequest***.

```
<SubscriptionRequest>
    <!--=======ENDPOINT REFERENCES================================-->
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <RequestorRef>NADER</RequestorRef>
    <!-- Subscription 1 for VIS123 to Place 975 -->
    <EstimatedTimetableSubscriptionRequest>
        <SubscriberRef>NADER</SubscriberRef>
```

```
        <SubscriptionIdentifier>000765</SubscriptionIdentifier>
        <InitialTerminationTime>2001-12-17T09:30:47-05:00</InitialTerminationTime>
        <EstimatedTimetableRequest version="1.0">
            <RequestTimestamp>2001-12-17T09:30:47-05:00</RequestTimestamp>
            <PreviewInterval>P1Y2M3DT10H30M</PreviewInterval>
            <TimetableVersionRef>0008</TimetableVersionRef>
            <OperatorRef>SMOOTH</OperatorRef>
            <LineRef>123</LineRef>
            <DirectionRef>INBOUND</DirectionRef>
        </EstimatedTimetableRequest>
        <ChangeBeforeUpdates>P1Y2M3DT10H30M</ChangeBeforeUpdates>
    </EstimatedTimetableSubscriptionRequest>
</SubscriptionRequest>
```

6.5 The EstimatedTimetableDelivery

6.5.1 Introduction

The ***EstimatedTimetableDelivery*** returns the predicted arrival times of a VEHICLE or group of VEHICLEs.

6.5.2 ServiceDelivery with a EstimatedTimetableDelivery — Element

One or more ***EstimatedTimetableSubscriptionRequest*** elements may be returned as part of a SIRI ***ServiceDelivery***.

Table 16 — ServiceDelivery / EstimatedTimetableDelivery — Attributes

ServiceDelivery			+Structure	See SIRI Part 2 ***ServiceDelivery***
HEADER	***:::***	**1:1**	See ServiceDelivery	
Payload	***EstimatedTimetableDelivery***	0:*	+Structure	See ***EstimatedTimetableDelivery*** element.

6.5.3 EstimatedTimetableDelivery — Element

Each ***EstimatedTimetableDelivery*** is made up of ***EstimatedVehicleJourney*** elements. There will be status messages for any requests that could not be fulfilled.

Table 17 — EstimatedTimetableDelivery — Attributes

EstimatedTimetableDelivery			*+Structure*	Describes one or more Dated Timetables.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Production timetable Service. Fixed.
LEADER	***:::***	**1:1**	xxx*ServiceDelivery*	Inherits from xxx***ServiceDelivery:*** See SIRI Part 2.
Payload	***EstimatedTimetable-VersionFrame***	0:*	*+Structure*	See ***EstimatedTimetableVersionFrame*** element.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

6.5.4 EstimatedTimetableVersionFrame — Element

Each production timetable is returned as an ***EstimatedTimetableVersionFrame*** element. Each ***Estimated-TimetableVersionFrame*** comprises one or more ***EstimatedVehicleJourney*** elements grouped for a version of the timetable.

Table 18 — EstimatedTimetableVersionFrame — Attributes

EstimatedTimetableVersionFrame			*+Structure*	Provides a schedule of DATED VEHICLE JOURNEY for a LINE and DIRECTION
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded.
Identity	***VersionRef***	0:1	*→VersionCode*	Reference to a TIMETABLE VERSION FRAME.
Journeys	***EstimatedVehicleJourney***	**1:***	*+Structure*	Provides real-time information about a VEHICLE JOURNEY along which a VEHICLE is running. See ***EstimatedVehicleJourney*** element.
Connections	***EstimatedServiceJourneyInterchange***	**0:***	*+Structure*	Connection parameters for a monitored SERVICE JOURNEY INTERCHANGE between a feeder and distributor journey. SIRI 2.0
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

6.5.5 EstimatedVehicleJourney — Element

6.5.5.1 Overview

Each ***EstimatedVehicleJourney*** contains an ordered list of ***EstimatedCall*** elements, as well as other properties.

There are three different ways to match the journeys described in the real-time data to the planned schedule:

— A full DatedVehicleJourneyRef, if known.

— A signature of the journey comprising its origin and destination points and arrival and departure times at those points.

— A real-time VEHICLE JOURNEY code (***EstimatedVehicleJourneyCode***).

Method (ii) allows installations that do not have a Production Timetable service still to make use of data.

Table 19 — EstimatedVehicleJourney — Attributes

EstimatedVehicleJourney				*+Structure*	Provides real-time information about the VEHICLE JOURNEY along which a VEHICLE is running.
Log	***RecordedAtTime***		**1:1**	*xsd:dateTime*	Time at which data for individual journey was recorded if different from that .on frame. SIRI 2.0
Vehicle Journey Identity	***LineRef***		**1:1**	*→LineCode*	Reference to the LINE of the VEHICLE JOURNEY.
	DirectionRef		**1:1**	*→DirectionCode*	Reference to the relative DIRECTION the VEHICLE is running along the LINE, for example, "in" or "out", “clockwise”. Distinct from a destination.
				choice	One of the following three ways of identifying a journey.
	a	**FramedVehicle-JourneyRef**	**1:1**	*+Structure*	Reference to the VEHICLE JOURNEY. The Frame is pvided by the EstimatedTimetableVersionFrame that contains the journey.
	b	***DatedVehicle-Journey-IndirectRef***		*+Structure*	If no VEHICLE JOURNEY reference is available, identify it by origin and destination and the scheduled times at these stops.
		OriginRef	**1:1**	*→StopPointCode*	The origin is used to help identify the VEHICLE JOURNEY.
		Aimed-Departure-Time	**1:1**	*xsd:dateTime*	Departure time at origin stop.
		Destination-Ref	**1:1**	*→StopPointCode*	The origin is used to help identify the VEHICLE JOURNEY.
		AimedArrival-Time	**1:1**	*xsd:dateTime*	Arrival time at origin stop.
	c	***EstimatedVehicle JourneyCode***	**1:1**	*Estimated-VehicleJourney-Code*	If this is the first message about an extra unplanned VEHICLE JOURNEY, a new and unique code shall be given for it. ***ExtraJourney*** should be set to *'true'*.
Change	***ExtraJourney***		0:1	*xsd:boolean*	Whether this journey is an addition to the plan. Can only be used when both participants recognise the same schedule version. If omitted, defaults to *false*: the journey is not an addition.
	Cancellation		0:1	*xsd:boolean*	Whether this journey is a cancellation of a journey in the plan. Can only be used when both participants recognise the same schedule version. If omitted, defaults to *false*: Journey is not a cancellation.
Journey Pattern Info	***:::***		0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Service Info	***:::***		0:1	*ServiceInfoGroup*	See SIRI Part 2 ***ServiceInfoGroup***.
Journey Info Group	***:::***		0:1	*JourneyInfoGroup* See above	See SIRI Part 2 ***JourneyInfoGroup.***
Estim-ated Info	***HeadwayService***		0:1	*xsd:boolean*	Whether this is a Headway Service, that is one shown as operating at a prescribed interval rather than to a fixed timetable.
	FirstOrLastJourney		0:1	*FirstOrLastJourneyEnum*	Whether journey is first or last journey of day. +SIRI v2.0

Disruption-Group	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***DisruptionGroup***. *DetailLevel*: normal.
Real-time Info	***Monitored***	0:1	*xsd:boolean*	Whether there is real-time information available for journey, if not present, not known. Inheritable.
	PredictionInaccurate	0:1	*xsd:boolean*	Whether the VEHICLE is in congestion. If not, present, not known. Inheritable.
	Occupancy	0:1	*OccupancyEnum*	How full the VEHICLE is. If omitted: passenger load is unknown. See Part2 Common definitions *full* | *seatsAvailable* | *standingAvailable*
Operational Info	***:::***	0:1	*OperationalInfo-Group*	See SIRI Part 2 ***OperationalInfoGroup.***
Calls	***RecordedCalls***	0:1	*+Structure*	Complete sequence of stops already visited along the route path, in calling order. Only used if observed stop data is being recorded. (SIRI 2.0)
	RecordedCall	1:*	*+Structure*	Individual ***Recorded Call***. See below. Only used if observed stop data is being recorded. (SIRI 2.0)
	EstimatedCalls	0:1	*+Structure*	Complete sequence of stops along the route path, in calling order. Normally this is only the onwards stops from the vehicle's current position.
	EstimatedCall	1:*:*	*+Structure*	Individual ***EstimatedCall***. See below.
	IsComplete-StopSequence	0:1	*xsd:boolean*	Whether the above CALL sequence is complete, i.e. represents every CALL of the route and so can be used to replace a previous CALL sequence Default is *'false'*.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

6.5.5.2 RecordedCall — Element (SIRI 2.0)

Each ***RecordedCall*** describes the observed times at a stop that has already been visited.

Table 20 — RecordedCall — Attributes

RecordedCall			*+Structure*	Provides real-time information about a CALL at a SCHEDULED STP POINT..
Stop Identity	***StopPointRef***	1:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT.
	VisitNumber	0:1	*VisitNumber-Type*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to a stop, the ***VisitNumber*** count is used to distinguish each separate visit. If not specified, Default is *'1'*.
	Order	0:1	*xsd:positive-Integer*	For implementations for which the overall order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Name of stop. One per language (Unbounded 0:* since +SIRI v2.0).
Change	***ExtraCall***	0:1	*xsd:boolean*	Whether this CALL was an addition to the plan.
	Cancellation	0:1	*xsd:boolean*	Whether this CALL was a cancellation of a CALL in the plan

Real time Info	***Prediction-Inaccurate***	0:1	*xsd:boolean*	Whether the prediction was flagged as being inaccurate because of congestion.
	Occupancy	0:1	*OccupancyEnum*	How full the VEHICLE was at the stop. Enumeration. If omitted: ***Occupancy*** is as for journey. Enumeration. See Part2 Common definitions. *full* | *seatsAvailable* | *standingAvailable*
Arrival	***AimedArrival-Time***	0:1	*xsd:dateTime*	Planned arrival time in either the original or Production Timetable. Can be omitted if equal to the departure time, or if the time is for the VEHICLE at the origin stop.
	ExpectedArrival-Time	0:1	*xsd:dateTime*	Predicted time of arrival of VEHICLE at time of record. If different from aimed arrival time
	ActualArrival-Time	0:1	*xsd:dateTime*	Recorded time of arrival of VEHICLE.
	Arrival-PlatformName	0:1	*NLString*	Bay or platform (QUAY) name at which VEHICLE actually arrived if different from departure platform..
Departure	***AimedDeparture-Time***	0:1	*xsd:dateTime*	Planned departure time in either the original or Production Timetable. Can be omitted if equal to the departure time, or if the time is for the VEHICLE at the origin stop.
	Expected-DepartureTime	0:1	*xsd:dateTime*	Predicted time of departure of VEHICLE at time of record.
	ActualDeparture-Time	0:1	*xsd:dateTime*	Recorded time of departure of VEHICLE.
	Departure-PlatformName	0:1	*NLString*	Bay or platform (QUAY) name from which VEHICLE actually departef.
Headway Interval Group	***AimedHeadway-Interval***	0:1	*PositiveDuration*	Target interval for frequency based service.
	Expected-HeadwayInterval	0:1	*PositiveDuration*	Expected interval for frequency based service.
	ActualHeadway-Interval	0:1	*PositiveDuration*	Recorded interval for frequency based service.
Stop Proximity Group	***DistanceFrom-Stop***	0:1	*DistanceType*	Distance of VEHICLE from stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. Positive value denotes distance before stop. +SIRI v2.0.
	NumberOf-StopsAway	0:1	*nonNegative-Integer*	Count of stops along SERVICE PATTERN between current position of VEHICLE and stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. +SIRI v2.0.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

6.5.5.3 EstimatedCall — Element

Each ***EstimatedCall*** describes the estimated future times at a stop. A journey shall contain at least two CALLs of any type.

NOTE If the ***Occupancy*** field in the ***EstimatedCall*** structure is filled, this represents a predicted passenger load. If the corresponding field is filled in a ***MonitoredVehicleJourney***, this overwrites the last current passenger occupancy value.

Table 21 — EstimatedCall — Attributes

EstimatedCall			*+Structure*	Provides real-time information about a CALL at a stop.
Stop Identity	***StopPointRef***	1:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT.
	VisitNumber	0:1	*VisitNumber-Type*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to a stop, the ***VisitNumber*** count is used to distinguish each separate visit. If not specified, Default is *'1'*.
	Order	0:1	*xsd:positive-Integer*	For implementations for which the overall order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Name of stop. One per language (Unbounded 0:* since +SIRI v2.0).
Change	***ExtraCall***	0:1	*xsd:boolean*	Whether this CALL is an addition to the plan. Can only be used when both participants recognise the same schedule version. If omitted, defaults to *false*: the CALL is not an addition.
	Cancellation	0:1	*xsd:boolean*	Whether this CALL is a cancellation of a CALL in the plan. Can only be used when both participants recognise the same schedule version. If omitted, defaults to *false*: CALL is not a Cancellation.
Real time Info	***Prediction-Inaccurate***	0:1	*xsd:boolean*	Whether the VEHICLE is in congestion. If not, present, not known. Inheritable.
	Occupancy	0:1	*OccupancyEnum*	How full the VEHICLE is at the stop. Enumeration. If omitted: ***Occupancy*** is as for journey. Enumeration. See Part2 Common definitions. *full* | *seatsAvailable* | *standingAvailable*
Call Property	***TimingPoint***	0:1	*xsd:boolean*	Whether the stop is a TIMING POINT. Times for stops that are not TIMING POINTs are sometimes interpolated crudely from the TIMING POINTs, and may represent a lower level of accuracy. Default is *'true'*.
	BoardingStretch	0:1	*xsd:boolean*	Whether this is a Hail and Ride stop. Default is *'false'*.
	RequestStop	0:1	*xsd:boolean*	Whether VEHICLE stops only if requested explicitly by passenger. Default is *'false'*.
	OriginDisplay	0:*	*NLString*	The name of the origin of the VEHICLE JOURNEY; used to help identify the VEHICLE to the public: Since VEHICLEs can change their displayed origin during a journey, the origin included here should be what the VEHICLE will display as it reaches this stop. (SIRI v2.0). One per language
	Destination-Display	0:*	*NLString*	The name of the destination of the VEHICLE JOURNEY; used to help identify the VEHICLE to the public: Since VEHICLEs can change their destination during a journey, the destination included here should be what the VEHICLE will display when it reaches this stop. One per language (Unbounded 0:* since +SIRI v2.0).
Note	***CallNote***	0:*	*NLString*	Text annotation that applies to this CALL.
Disruption Group	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***DisruptionGroup***. *DetailLevel*: normal.

Arrival	***AimedArrival-Time***	0:1	*xsd:dateTime*	Planned arrival time in either the original or Production Timetable. Can be omitted if equal to the departure time, or if the time is for the VEHICLE at the origin stop.
	ExpectedArrival-Time	0:1	*xsd:dateTime*	Estimated time of arrival of VEHICLE.
	:::	0:1	*MonitoredStop-Arrival StatusGroup*	Set of states description elements for *arrival* ***Monitored-StopArrivalStatusGroup***. See later below.
Departure	***:::***	0:1	*Onwards Departure TimesGroup*	See later below ***OnwardsVehicleDepartureTimes-Group***. (part of ***MonitoredVehicleJourney***).
	:::	0:1	*PassengerDepar tureTimesGroup*	See later below
	:::	0:1	*MonitoredStop DepartureStatus-Group.*	Set of states description elements for arrival. *See* ***MonitoredStopDepartureStatusGroup*** later below . (part of ***MonitoredVehicleJourney***).
Headway Interval Group	***AimedHeadway-Interval***	0:1	*PositiveDuration*	Target interval for frequency based service.
	Expected-HeadwayInterval	0:1	*PositiveDuration*	Estimated interval for frequency based service.
Stop Proximity Group	***DistanceFrom-Stop***	0:1	*DistanceType*	Distance of VEHICLE from stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. Positive value denotes distance before stop. +SIRI v2.0.
	NumberOf-StopsAway	0:1	*nonNegative-Integer*	Count of stops along SERVICE PATTERN between current position of VEHICLE and stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. +SIRI v2.0.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

6.5.6 EstimatedServiceJourneyInterchange — Element (SIRI v2.0)

Overview

Each ***EstimatedServiceJourneyInterchange*** represents a real-time SERVICE JOURNEY INTERCHANGE that may be made at a specified point. It includes the current real time predicted wait values.

Table 22 — EstimatedServiceJourneyInterchange — Attributes

EstimatedServiceJourneyInterchange				*+Structure*	Provides real-time information about a SERVICE JOURNEY INTERCHANGE that is being monitored. +SIRI v2.0
Connecting Vehicle Journey Identity	***InterchangeRef***		**0:1**	*→Interchange-Code*	Reference to an interchange at which connection takes place. +SIRI v2.0
	FeederJourneyRef		**1:1**	*+Structure*	Reference to a connecting feeder VEHICLE JOURNEY. +SIRI v2.0
		FramedVehicle-JourneyRef	**1:1**	*+Structure*	Frame qualified Reference to a connecting feeder VEHICLE JOURNEY. +SIRI v2.0
			DataFrame-Ref	**0:1**	*→Vehicle-JourneyCode*	Reference to a data frame within participant service. Used to ensure that the ***DatedVehicleJourneyRef*** is unique with the data horizon of the service. Often the ***OperationalDayType*** is used for this purpose. +SIRI v2.0
			DatedVehicle-JourneyRef	**0:1**	*→DatedVehicle-JourneyCode*	A reference to the DATED VEHICLE JOURNEY that the VEHICLE is making. +SIRI v2.0
		LineRef		**0:1**	*➔LineCode*	A reference to the LINE of the referenced feeder VEHICLE JOURNEY. +SIRI v2.0
		ParticipantRef		**0:1**	*➔ ParticipantCode*	A reference to the system from which the feeder reference comes
	Distributor-JourneyRef			**1:1**	*+Structure*	Reference to a connecting distributor VEHICLE JOURNEY. +SIRI v2.0
		FramedVehicle-JourneyRef		**1:1**	*+Structure*	Frame qualified Reference to a connecting distributor VEHICLE JOURNEY. +SIRI v2.0
			DataFrame-Ref	**0:1**	*→Vehicle-JourneyCode*	Reference to a data frame within participant service. Used to ensure that the distributor DatedVehicleJourneyRef is Unique with the data horizon of the service. Often the OperationalDayType is used for this purpose. +SIRI v2.0
			DatedVehicle-JourneyRef	**0:1**	*→DatedVehicle-JourneyCode*	A reference to the distributor DATED VEHICLE JOURNEY that the VEHICLE is making. +SIRI v2.0
		LineRef		**0:1**	*➔LineCode*	A reference to the LINE of the referenced distributor VEHICLE JOURNEY. +SIRI v2.0
		ParticipantRef		**0:1**	*➔ ParticipantCode*	A reference to the system from which the distributor reference comes
Interchange parameters					*choice*	One of the following three ways of identifying a journey.
	a	***WillNotWait***		**–**	*EmptyType*	Distributor will not wait (i.e. connection broken) +SIRI v2.0
	b	***WillWait***		**1:1**	*+Structure*	If no VEHICLE JOURNEY reference is available, identify it by origin and destination and the scheduled times at these stops. +SIRI v2.0
			WaitUntilTime	**1:1**	*xsd:dateTime*	Time up until which the distributor will wait. SIRI v2.0
			DriverHas-AcknowledgedWIllWait	0:1	*xsd:boolean*	Whether an acknowledgement has been received that the driver will wait. SIRI v2.0
	ExpectedDeparture-TimeOfDistributor			0:1	*xsd:dateTime*	Time at which distributor VEHICLE is expected to depart. SIRI v2.0
	Connection-Monitoring			0:1	*xsd:boolean*	Whether connection monitoring is active or not for this connection SIRI v2.0.

6.5.7 EstimatedTimetableDelivery — Example

The following is an example of an ***EstimatedTimetableDelivery***.

```
<ServiceDelivery>
    <!--=======HEADER================================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <MoreData>false</MoreData>
    <!--=======PAYLOAD================================================ -->
    <EstimatedTimetableDelivery version="1.0">
        <ResponseTimestamp>2001-12-17T09:30:47-05:00</ResponseTimestamp>
        <SubscriberRef>NADER</SubscriberRef>
        <SubscriptionRef>0004</SubscriptionRef>
        <Status>true</Status>
        <ValidUntil>2001-12-17T19:30:47-05:00</ValidUntil>
        <EstimatedVehicleJourney>
            <LineRef>LZ123</LineRef>
            <DirectionRef>INBOUND</DirectionRef>
            <DatedVehicleJourneyRef>00008</DatedVehicleJourneyRef>
            <Cancellation>false</Cancellation>
            <PublishedLineName>124</PublishedLineName>
            <OperatorRef>SMOOTH</OperatorRef>
            <ProductCategoryRef>NMTOKEN</ProductCategoryRef>
            <ServiceFeatureRef>NMTOKEN</ServiceFeatureRef>
            <VehicleFeatureRef>DisabledAccess</VehicleFeatureRef>
            <ServiceFeatureRef>CyclesPermitted</ServiceFeatureRef>
            <DestinationName>Howard's End</DestinationName>
            <VehicleJourneyName>Shoppers Special</VehicleJourneyName>
            <JourneyNote>Not o bank holidays</JourneyNote>
            <Monitored>true</Monitored>
            <PredictionInaccurate>false</PredictionInaccurate>
            <PassengerLoad>full</PassengerLoad>
            <!-- Call 1-->
            <EstimatedCall>
                <StopPointRef>00001</StopPointRef>
                <ExtraStop>false</ExtraStop>
                <PredictionInaccurate>false</PredictionInaccurate>
                <Occupancy>SeatsAvailable</Occupancy>
                <BoardingStretch>false</BoardingStretch>
                <RequestStop>false</RequestStop>
                <CallNote>Starts here</CallNote>
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
                <ArrivalBoardingActivity>NoAlighting</ArrivalBoardingActivity>
                <AimedDepartureTime>2001-12-17T09:30:47-05:00</AimedDepartureTime>
                <DeparturePlatformName>1</DeparturePlatformName>
            </EstimatedCall>
            <!-- Call 2-->
            <EstimatedCall>
                <StopPointRef>00002</StopPointRef>
                <ExtraStop>false</ExtraStop>
                <PredictionInaccurate>false</PredictionInaccurate>
                <Occupancy>SeatsAvailable</Occupancy>
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
                <AimedDepartureTime>2001-12-17T09:30:47-05:00</AimedDepartureTime>
                <ExpectedArrivalTime>2001-12-17T09:30:47-05:00</ExpectedArrivalTime>
                <ExpectedDepartureTime>2001-12-17T09:30:47-05:00</ExpectedDepartureTime>
                <BoardingActivity>BoardingAndAlighting</BoardingActivity>
                <RequestStop>true</RequestStop>
                <DeparturePlatformName>3</DeparturePlatformName>
                <ArrivalPlatformName>4</ArrivalPlatformName>
            </EstimatedCall>
            <!-- Call 3 -->
            <EstimatedCall>
                <StopPointRef>00003</StopPointRef>
                <ExtraStop>false</ExtraStop>
                <PredictionInaccurate>true</PredictionInaccurate>
```

```
                <Occupancy>full</Occupancy>
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
                <ExpectedArrivalTime>2001-12-17T09:30:47-05:00</ExpectedArrivalTime>
                <ArrivalPlatformName>5</ArrivalPlatformName>
                <AimedDepartureTime>2001-12-17T09:30:47-05:00</AimedDepartureTime>
                <ExpectedDepartureTime>2001-12-17T09:30:47-05:00</ExpectedDepartureTime>
                <DepartureBoardingActivity>NoBoarding</DepartureBoardingActivity>
            </EstimatedCall>
            <IsCompleteStopSequence>false</IsCompleteStopSequence>
        </EstimatedVehicleJourney>
    </EstimatedTimetableDelivery>
</ServiceDelivery>
```

6.6 Handling of Predictions in the Estimated Timetable Service

6.6.1 Supplementary Rule for the Delay Profile

Various mechanisms are supported in SIRI to prevent unnecessary data traffic. Some of these can involve a limited amount of independent recomputation by the schedule information system to process the transmitted data, in particular delays; this recomputation should be done in a consistent manner.

In particular, to reduce the transmitted data volume, the producer *only need complete the stops at which the delay has changed* (the 'interpolation rule'). The schedule information system assigns the last reported delay (or rather schedule deviation, as a VEHICLE may also be early) to all stops along the route until the next reported delay is received.

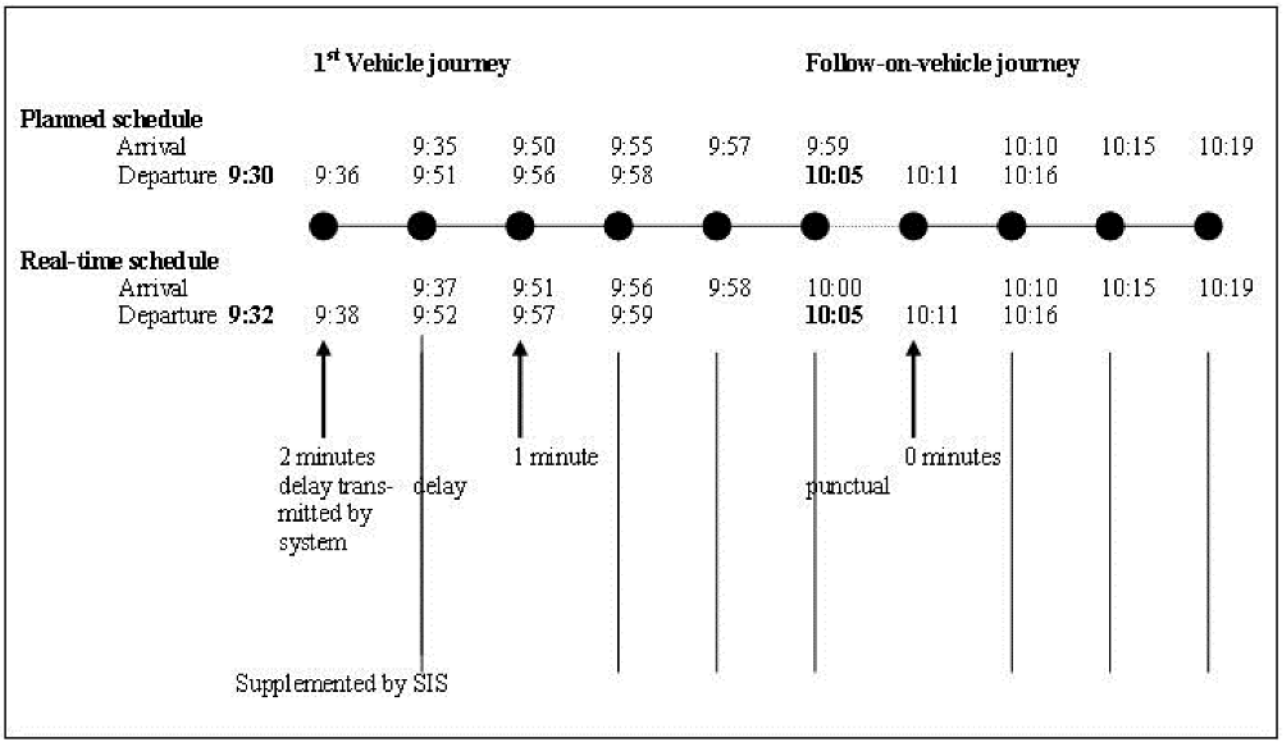

Figure 1 — Supplementary Rule for Delay Extrapolation

The Producer can use its authority to adjust predictions if it establishes that there are waiting times at upcoming stops that could compensate for the delay. The schedule information system receives a message for these stops by means of multiple ***EstimatedCall*** structures (in a single delivery). For the stops in between, the schedule information system shall interpolate the current real-time status. In this case the overall sovereignty for making delay prognoses still rests with the Producer and the principle that it is the authoritative

source is not violated. Using interpolation, the schedule information system may then complete the remaining fields from a single transmission, without needing additional information from subsequent or previous messages.

6.6.2 Other Supplementary Rules

Interpolation rules shall be agreed for each implementation. The use of the interpolation rule is only recommended for the transmission of delays or early times.

On the basis of early practical experiences with inter-system AVMS interfaces, further rules should be exactly specified for the independent supplementation of transmitted data in order to avoid any confusion in interpretation. The producer control computer shall know the stops for which the schedule information system carries out extrapolation and the stops for which the predictions need to be sent. In SIRI an indication can be given as to whether a stop is a timing point or not which may be relevant to the interpolation process.

6.6.3 Quality of Prognoses and Prediction Windows

Customers expect VEHICLE JOURNEY information from real-time enabled systems to be more accurate, informative and timely than the planned schedules. For example, with the help of the SIRI Estimated Timetable service the schedule information system can mark VEHICLE JOURNEYs as “delayed x min”, “punctual” or “cancelled”.

Systems should therefore be able to distinguish between there being no messages from the Producer because there is no significant delay, from the case when the Producer is unable to generate any messages for the VEHICLE at all, for example, because the VEHICLE has no radio equipment. In the first case the VEHICLE JOURNEY should be explicitly marked as punctual in the schedule information system; and in the latter case, as having an unknown accuracy. SIRI includes a number of elements (***Monitored***, ***PredictionInaccurate***, ***InCongestion***, ***InPanic***) that may be used to represent this.

In order to be able to make reliable predictions, the schedule information system should base its VEHICLE JOURNEY information on information from explicit messages, and not on conclusions drawn from the assumed successful passing of events.

The further ahead in time a prediction is being attempted, the less certain factors will influence it, and so the greater likelihood of inaccuracy. This increasing indeterminism also applies to the prediction of traffic flows. In practice, predictions are only reliable for VEHICLE JOURNEYs in the near future. To avoid misrepresenting the apparent accuracy of prognoses, every producer system has a restricted ***PreviewInterval***, which defines a rolling forward window within which predictions will be attempted. The Preview Interval is specific to each implementation and lies in the region of 20-60 min. This prevents it attempting to make predictions about the punctuality of VEHICLE JOURNEYs that lie too far in the future.

Passengers do not in any case expect an exact long-term prognosis of punctuality. It would make little sense then for a schedule information system, when responding to an enquiry made in the morning about a journey that evening, to mark the evening VEHICLE JOURNEYs as ‘punctual’.

When setting up a SIRI Estimated Timetable subscription, the schedule information system requests a desired Preview Interval from the Producer.

6.6.4 Reporting Rules and Monitored Journeys

The Producer shall report every VEHICLE JOURNEY for which there are predictions at least once.

The Producer only transmits the predictions for the VEHICLE JOURNEYs that start within the Preview Interval. A future VEHICLE JOURNEY lies within the Preview Interval if the departure time of the VEHICLE JOURNEY at the origin SCHEDULED STOP POINT lies in the time window from the current time until the end time given by the Preview Interval.

The prediction message shall be generated as early as possible within the Preview Interval, that is, as soon as the Producer can give a prediction for the start of the VEHICLE JOURNEY. If this is not possible, a message shall be generated for the VEHICLE JOURNEY after the comparison between the planned and real-time data (after logon, after first radio contact, before/at/after the start stop).

As soon as a follow-on VEHICLE JOURNEY falls into the Preview time, it too shall be reported with its entire delay profile as soon as possible (i.e. with the prediction for the start stop and all subsequent stops).

For every transmitted VEHICLE JOURNEY, the predicted delay profile is reported up to the end of the VEHICLE JOURNEY, even if the VEHICLE JOURNEY finishes after the prediction horizon.

A VEHICLE JOURNEY that has been reported once shall continue to be reported until it is completed, or until predictions become impossible.

The schedule information system can therefore assume a VEHICLE JOURNEY to be punctual if it has received an initial message for the journey from the Producer, and if the ***Monitored*** property has not been set to false. Without the active transmission of this information, the journey is regarded as unmonitored, and the schedule information system switches back to using only planning data for the journey.

This procedure ensures that the schedule information system only marks VEHICLE JOURNEYs as punctual when they are actually monitored by the Producer, and are being transmitted without error. It is possible to further improve the reliability if, in the case of punctual vehicles too, the Producer continues to send messages from time to time after the first real-time message.

The disadvantage is the slightly greater volume of data that needs to be transmitted. The re-initialisation of a real-time data subscription (e.g. after a restart) in particular would generate a new message to the schedule information system from every VEHICLE affected by the subscription currently in operation.

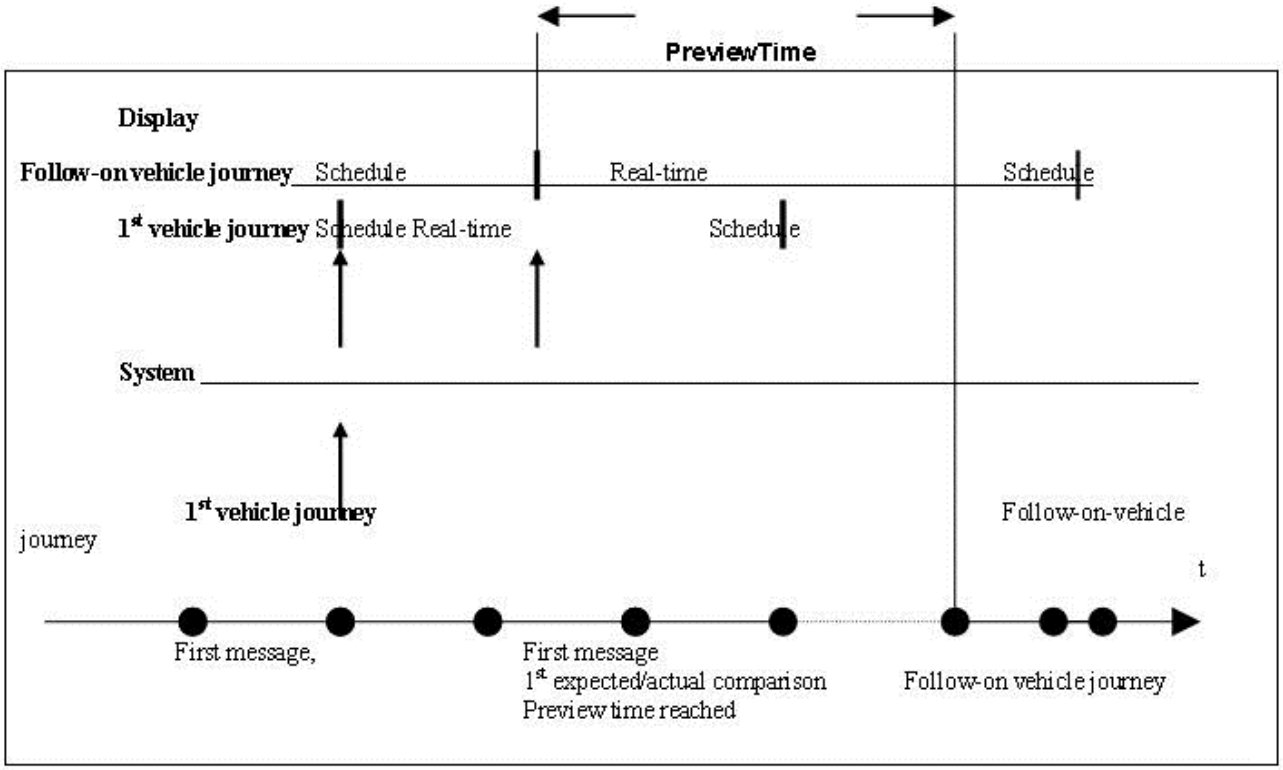

Figure 2 — Preview Time and Active First Message

6.6.5 Temporal Reporting Behaviour – Sensitivity Threshold

The SIRI Estimated Timetable service reports the current absolute arrival and departure times for a VEHICLE JOURNEY. The content of a delay message therefore does not represent information as a difference to a value in the preceding message, e.g. “+5 min”, or to the static planned schedule. Every individual SIRI Estimated Timetable message can be interpreted on its own. It can be made available in the schedule information system if there is a corresponding reference to the planned data.

Stop Visits and monitored journeys may also be assigned a descriptive categorization of their relative progress; for example ‘onTime’, ‘late’, ‘early’: this progress status is a presentation hint to display systems that may reflect an informational policy of the operator, for example that trains less than two minutes behind the planned times should be regarded as on-time regardless.

The triggering of the communication of VEHICLE JOURNEY delays however is relative to differences in the data since the last message, and takes the form of a hysteresis function: As soon as a delay prediction for a stop exceeds the last transmitted value by a specific threshold (either negatively or positively), the Producer triggers a real-time message to the schedule information system which overwrites the old value.

It is recommended that a single sensitivity threshold be globally defined for the entire real-time data subscription, with a value in the range of one to two minutes. The alternative approach of defining the threshold values for each LINE, or even each VEHICLE JOURNEY or interval is not recommended because of the resulting complexity. Instead, the presence of a global sensitivity value should be interpreted as signifying that the schedule information system does not wish to receive any messages whose difference in content lies below the threshold value. For transmissions in the opposite direction however, the Producer may suppress messages that lie above the threshold as it is in a better position to determine whether it would be useful to transmit the data or not. For example, the messages could be suppressed when the headways are short, i.e. the VEHICLEs run in rapid succession. This allows the Producer to dynamically vary threshold values as appropriate.

A preview interval can be used to limit the data horizon for predictions. Other changes, for example additional journeys or journey cancellations, will still be included even if beyond the prediction horizon.

6.6.6 Prediction Inaccurate – In Congestion

If the AVMS establishes that a VEHICLE is travelling extremely slowly or is at a complete standstill, it can activate the ***InCongestion*** and ***PredictionInaccurate*** elements in the ***JourneyProgressGroup*** structure for the relevant VEHICLE JOURNEY. At the same time, the operation of hysteresis mentioned above is suspended, i.e. the Producer suppresses further messages as long as ***PredictionInaccurate*** is active. This avoids the continuing exchange of redundant messages that would arise as the delay increased. The system avoids sending delay messages when it shall be assumed they will only increase further after another 2 min.

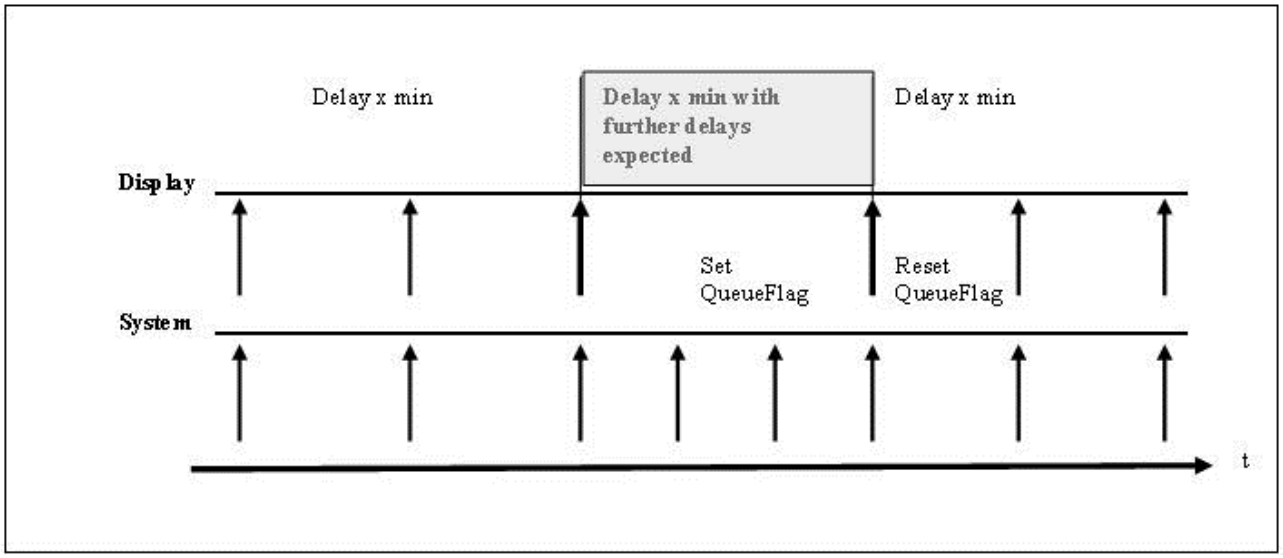

Figure 3 — Reporting Behaviour in the case of PredictionInaccurate

With the transmission of a new current prediction, the Producer deactivates the ***PredictionInaccurate*** status and signals that the change sensitivity function has been resumed.

The schedule information system can provide special behaviour to process ***PredictionInaccurate*** content. In contrast to how it handles a breakdown in communication, the schedule information system does not revert to the backup level of planned data, but is able to trigger messages such as “20 min delay, more delays expected”.

The ***PredictionInaccurate*** is an overall status: it may be further explained by additional elements such as ***InCongestion*** and ***InPanic***.

6.6.7 Unexpected Termination of Monitoring

If a VEHICLE logs off from its block, or becomes unattainable via radio (‘loss of contact’), the Producer shall be able to revoke a previously reported prediction. In this case, for every VEHICLE JOURNEY already reported, the Producer sends the Consumer system an ***EstimatedVehicleJourney*** message with the ***Monitored*** attribute set to false (i.e. unmonitored). This action makes the schedule information system aware of the inaccuracy pertaining to predictions for these VEHICLE JOURNEYs, so that it can inform passengers. After a message reporting a VEHICLE JOURNEY as unmonitored, the journey has the same status as if it had not been reported at all. A more specific diagnosis of the error condition (e.g. GPRS, radio, etc.) can be included in the form of a ***MonitoringError*** code, allowing Producer systems to handle different failure modes differently.

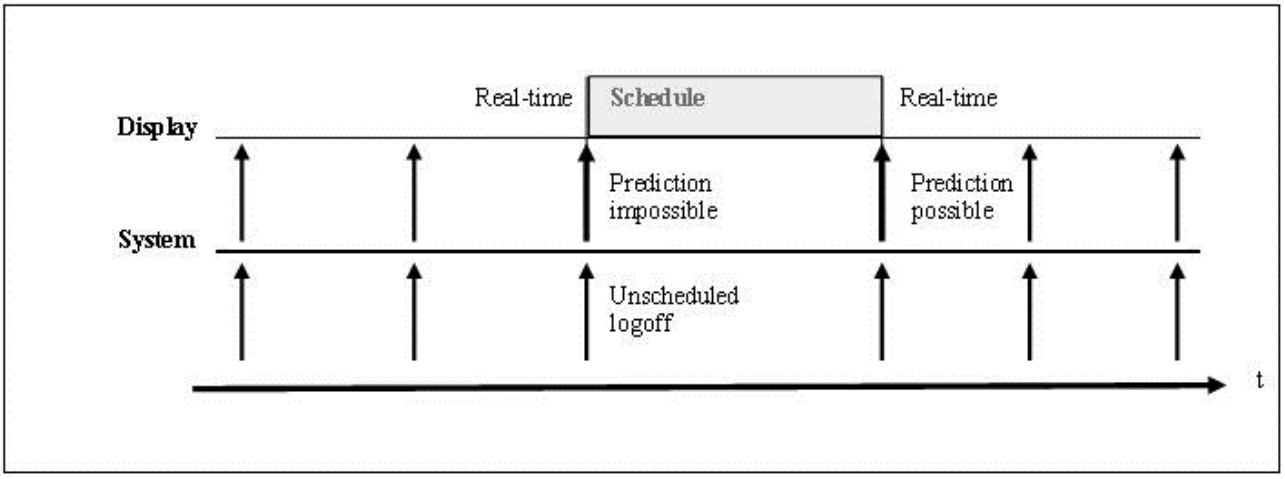

Figure 4 — Loss of Contact

6.7 Prediction Quality

6.7.1 General

In addition to predicted time values from the provider, a measure of prediction quality is useful for passenger information systems, especially in schedule based information systems, which handle real-time information. These systems can use such a prediction quality level together with the predicted time values to improve routing results for passengers.

6.7.2 Definition

Every predicted time value for a CALL (arrival and departure) may be augmented with a prediction quality level, which describes the level of reliability as the producing system (AVMS) may classify it. These are integer numerical values from 1 to 5, where 1 is reliable and 5 is unpredictable. Implicitly they define time intervals, in which the deviation of the predicted value from the observed value may fit with some high probability. The exact definition of each level of probability (a confidence level) itself is not part of the interface definition and has to be agreed between producer and consumer of the prediction and its quality factors. As a default, these intervals are defined relative to the predicted time value for every prediction level as follows:

Table 23 — Prediction Quality Levels

	Prediction quality level	**Lower Bound (earliest predictable arrival /departure time)**	**Upper Bound (latest predictable arrival /departure time)**	**Width of interval**
1	Very reliable	-1 min	+2 min	3 min
2	Reliable	-3 min	+6 min	9 min
3	Unreliable	-8 min	+16 min	24 min
4	Very unreliable	-20 min	+40 min	60 min
5	Prediction impossible	Undefined	Undefined	> 60 min

The suggested interval definitions are based on experience with real data gathered by long and short distance train operators, tram operators, city and overland bus operators and other schedule based public transport operators. Because delays often tend to grow, in practical situations, unsymmetrical intervals are used, which have one third before and two third after the predicted value.

Quality Level 5 has no related interval definition and will be used whenever no reliable quality of prediction is possible, i.e. the predicted value may spread more than 60 min.

The transmission of a certain level means that in the view of the producing system, the predicted value will fit into the related interval. This is in fact not a guaranteed property, but there are relatively simple statistical methods to test the agreed level of confidence a posteriori. It is due to the consumer system owner to agree with its different data providers about a comparable level of confidence.

6.7.3 Explicit prediction quality intervals

If for some reason known to the provider system (AVMS), the predefined intervals need to be modified for a given confidence level, specific revised interval bounds may be transmitted using the optional value pair **LowerTimeLimit, UpperTimeLimit.** If this feature is used, a consistency constraint needs upheld between the explicitly defined interval and the transmitted prediction level: the interval width relating to the transmitted prediction level according to the table above may not be smaller than the explicitly given interval width, which is defined as **UpperTimeLimit – LowerTimeLimit.** For example, the explicitly defined interval [-5, +5] may be provided with confidence level 3, 4 or 5, but not with 1 or 2. Otherwise the consuming system is free to downgrade the level for its internal use, especially in the routing procedure.

6.7.4 Extrapolation rule for prediction quality

Under the above definition, the prediction quality is not a general property of the whole journey, but rather local to a point in the journey and defines the prediction quality at a certain time for a specific arrival or departure schedule event. Therefore at a given time the prediction quality may differ between CALLs of the same journey, within the prediction time window.

To avoid excessive or redundant data traffic it is useful to define an extrapolation rule analogous to the delay propagation rule given in 6.6.1. If no information on prediction quality is available from the producer system for a given CALL, the consumer system may derive the prediction quality level from that of a previous CALL. This value may be propagated to all successive CALLs up to the next CALL for which an explicitly given level is provided (which may be higher or lower than the previous one).

If no prediction quality is given for any CALL of a journey (which is allowed from the XML syntax definition), no extrapolation should be made at all.

6.8 General Rules and Definitions for Timetable Data

6.8.1 Introduction

This clause outlines further important rules for the implementation and handling of the SIRI Production Timetable and SIRI Estimated Timetable services, including the handling of deleted, additional VEHICLE JOURNEYs, and the transmission of planned connections and connections where the passengers remain seated.

6.8.2 Cancelled Vehicle Journeys

There are two situations in which cancelled VEHICLE JOURNEYs needs to be communicated by the data producer:

— *Production Stage Cancellation:* the data producer and the consumer system both know the same version of the timetable; however, in the control centre a VEHICLE JOURNEY is cancelled before transmission of the planned schedules in the SIRI Production Timetable service. In this case, the VEHICLE JOURNEY to be cancelled in the planned schedule is communicated in a ***DatedTimetableVersionFrame****,* as a ***Dated-VehicleJourney*** marked as a ***Cancellation***.

— *Real-time Cancellation:* after completion of the SIRI Production Timetable transmission, a VEHICLE JOURNEY is cancelled in the control centre. In this case, the VEHICLE JOURNEY to be cancelled is communicated as an ***EstimatedVehicleJourney*** marked as a ***Cancellation***.

6.8.3 Additional Vehicle Journeys

There are two situations in which additional VEHICLE JOURNEYs shall be communicated by the data sender:

— *Production Stage Addition:* the data sender and the schedule information system both know the same version of the timetable; however, in the control centre, an additional VEHICLE JOURNEY is added before transmission of the planned schedules in the SIRI Production Timetable service. In this case, the additional VEHICLE JOURNEY is communicated in a ***DatedTimetableVersionFrame*** as a ***Dated-VehicleJourney*** marked with the ***ExtraJourney*** property.

— *Real-time Addition:* after completion of the SIRI Production Timetable transmission, an additional VEHICLE JOURNEY is added by the dispatcher. In this case, the additional VEHICLE JOURNEY is communicated as an ***EstimatedVehicleJourney*** marked with the ***ExtraJourney*** property.

If the additional VEHICLE JOURNEY follows a path that deviates from the other journeys of the LINE, it shall be guaranteed that this path has been previously supplied in the planning system. Otherwise, it may not be possible to represent the additional VEHICLE JOURNEY within the network topology of the schedule information system. In the case of an unknown path, there shall be a suitable error response from the schedule information system.

6.8.4 Changes to Journeys & Routings

In the case of major despatching alterations, it may be necessary to retransmit information for the entire route, including new planning and prognosis times. This is the case for any additional journeys not included in the planning data (see above), as well as for any JOURNEY PATTERN i.e. path changes to an existing route. The ***IsCompleteStopSequence*** on ***EstimatedVehicleJourney*** shall be set to “true” to inform the schedule information system that an entire route is being exchanged: if the path change involves the omission of stops when compared with the original plan, they are omitted altogether.

Changes to the attributes of a service, such as VEHICLE or service features (e.g. provision for bicycles, etc.) shall be sent as elements of the ***EstimatedVehicleJourney***.

6.8.5 Changes to Call Attributes

Changes to the attributes of a CALL at a stop, such as whether passengers can board or alight, shall be sent in an ***EstimatedCall***. If the CALL is cancelled altogether it shall be marked with the Cancellation property. The delay extrapolation rule cannot be applied in these situations.

6.8.6 Planned Connections and “Stay Seated” Connections

An interchange over a CONNECTION link exists between two VEHICLE JOURNEYs when a passenger can sensibly transfer between them as part of a meaningful journey.

Transfers between journeys at an interchange may be one-way: the feeder VEHICLE only sets down passengers, and the distributor only picks up, or two-way: that is, both VEHICLEs act as both feeder and distributor to each other, and if either is late for the rendezvous, the other will wait.

The interchange time needed to make a transfer may also be material. It would be inappropriate to link two VEHICLE JOURNEYs where the changeover time is not adequate to allow the passenger to reach the second VEHICLE.

Information as to which interchanges are possible at a CONNECTION link are possible through the course of the operating day under normal conditions (even where the changeover times are tight), is of great interest to

the schedule information system. For this reason, connecting VEHICLE JOURNEY pairs, along with the associated planning data, should be transferred to the schedule information system. The schedule information system represents the current viability of these transfers to the customer with display messages such as “connection expected”.

The planned connections are transmitted in the ***TargetedInterchange*** structure, which is part of a ***DatedCall***.

A “stay seated” transfer, where the passengers remain seated on the same VEHICLE, represents a special case. It is treated as a virtual connection in the same VEHICLE. This situation arises in practice when the VEHICLE in question transfers to a different LINE at the end of the current VEHICLE JOURNEY, hence to begin a new VEHICLE JOURNEY (LINE change within a block). For the passenger making a trip, the change between these two individual VEHICLE JOURNEYs is represented as a connection. However, the passenger can remain seated in the same VEHICLE (assuming this is permitted). In these circumstances, the information that she need not get off the VEHICLE is useful.

6.8.7 Handling Train Data

With train travel, it is often not possible to apply the regular LINE modelling as used in local public transport. However, in order to be able to apply the same data model for the train schedules, it is recommended that the timetable links for a corresponding LINE definition are used. In this case, the ***DatedVehicleJourneyRef*** will be the (unique) train number and the ***PublishedLineName*** will be composed of train type and train number (e.g. IC 18). This allows a train journey to be broken down into several “lines” on the basis of the timetable links.

7 Stop Timetable Service [ST]

7.1 Purpose

The SIRI Stop Timetable Service provides a stop-centric view of timetabled VEHICLE arrivals and departures at a designated stop. It can be used to reduce the amount of information that needs to be transmitted in real-time to stops and displays, as reference data for a Stop Monitoring Service; and provides a data feed of the static timetables.

The Stop Timetable Service comprises the ***StopTimetableRequest*** message to specify the information to be returned, and the ***StopTimetableDelivery*** message, used to deliver the response payload. The ***StopTimetableSubscription*** allows a subscriber to obtain asynchronous updates in the form of Stop Visits for a designated stop: it contains an embedded ***StopTimetableRequest,*** along with further parameters controlling the asynchronous delivery.

StopTimetableRequest has topic parameters to filter the information by stop, LINE, and DIRECTION, etc., and policy parameters to control the amount of data returned. The ***StopTimetableSubscription*** has further policy parameters, for example sensitivity threshold.

The ***StopTimetableDelivery*** returns information about one or more ***TimetabledStopVisit*** instance.

7.2 Reference Data

StopTimetableRequest and ***StopTimetableDelivery*** requires the participants to have agreed data reference models for :

a) Point (i.e. SCHEDULED STOP POINT, or Place);

b) LINE;

c) DIRECTION.

The ***StopPointsDiscoveryRequest*** can be used to obtain a list of stops. The ***LinesDiscoveryRequest*** can be used to obtain a list of LINEs, DIRECTIONs and destinations allowed in requests.

7.3 Capability and Permission Matrices

7.3.1 Capability Matrix

The following set of required and optional capabilities is defined for the Stop Timetable service. If the service supports Capability Discovery the ***StopTimetableCapabilitiesRequest*** / ***StopTimetable-CapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 24 — StopTimetableCapabilities Matrix

StopTimetableCapabilities				*+Structure*	Capabilities describing implementation of *StopTimetable* service.
inherit	***:::***		**1:1**	*See xxxCapability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***		0:1	*+Structure*	Which optional filtering features are supported?
		FilterByMonitoringRef	**1:1**	*xsd:boolean*	Whether results can be filtered by Monitoring reference. Required Capability: fixed as true.
		FilterByLineRef	**1:1**	*xsd:boolean*	Whether results can be filtered by ***LineRef***. Required Capability: fixed as true.
		FilterByDirectionRef	0:1	*xsd:boolean*	Whether results can be filtered by ***DirectionRef***, default is *'true'*.
Request Policy	***RequestPolicy***		0:1	*+Structure*	Which optional features of the Request Policy are supported by service?
		Language	**1:***	*xsd:language*	National languages used by service.
		Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
				choice	Location: one of the following two
	a	***GmlCoordinateFormat***	0:1	*SrsNameType*	Default coordinate format is given by a GML value.
	b	***WgsDecimalDegrees***		*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
		UseReferences	0:1	*xsd:boolean*	Whether results can return references for stops. Default is *'true'*.
		UseNames	0:1	*xsd:boolean*	Whether results can return names for stop. Default is *'true'*.
Access Control	***AccessControl***		0:1	*+Structure*	Which optional Access Control features are supported by service?
		RequestChecking	0:1	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
		CheckOperatorRef	0:1	*xsd:boolean*	If access control is supported, whether access control by OPERATOR is supported. Default is *'true'*.
		CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
		CheckMonitoringRef	0:1	*xsd:boolean*	If access control is supported, whether access control by monitoring reference (LOGICAL DISPLAY) is supported. Default is *'true'*.
any	***Extensions***		0:1	*xsd:any**	Placeholder for user extensions.

7.3.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***StopTimetableCapabilitiesResponse*** can include the access permissions for the requestor participant to access data.

Table 25 — Stop Timetable Service — Permissions

StopTimetablePermission			*+Structure*	Permissions to use implementation of Stop Timetable service.
In-herit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant. See Part 2.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant. See Part 2.
	StopMonitorPermissions	0:1	*+Structure*	Monitoring Reference (LOGICAL DISPLAY) permissions for participant. See Part 2.

7.4 StopTimetableRequest

7.4.1 StopTimetableRequest — Element

The ***StopTimetableRequest*** can be used in both a direct request, and for a subscription. If used for a subscription, additional ***StopTimetableSubscriptionPolicy*** parameters can be specified. Only a single stop or display point may be specified in a given ***StopTimetableSubscription***; a ***SubscriptionRequest*** may contain multiple service subscriptions per request, one for each stop. In this case, a single ***ServiceDelivery*** will be returned containing separate ***StopTimetableDelivery*** messages for each SCHEDULED STOP POINT.

The primary *Topic* terms on the request are the ***DepartureWindow*** which specifies a time window for selecting data, and the ***MonitoringRef***, which identifies the stop or display for which information is to be returned. Additional topic filtering on LINE and DIRECTION is also allowed.

Table 26 — StopTimetableRequest — Attributes

StopTimetableRequest			*+Structure*	Request information about arrival/departure visits at a stop.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of *StopTimetable* Service, e.g. 1.0c.
Endpoint Properties	***Request-Timestamp***	**1:1**	*xsd:dateTime*	See Part 2.
	MessageIdentifier	0:1	*MessageQualifier*	
Topic	***DepartureWindow***	0:1	*ClosedTime-Interval*	Time frame for which data is to be supplied. The reference is the departure time of the feeder at the CONNECTION link. If absent use data horizon of server system.
	MonitoringRef	**1:1**	*→Monitoring-Code*	The Stop Monitoring point for which Stop Visits will be returned. May be an identifier of a SCHEDULED STOP POINT or a display.
	LineRef	0:1	*→LineCode*	Filter the results to include only Stop Visits for the given LINE
	DirectionRef	0:1	*→DirectionCode*	Filter results to include only Stop Visits for VEHICLEs running in a specific relative DIRECTION, for example, "inbound" or "outbound". (Direction does not specify a destination.) Optional SIRI capability: *TopicFiltering* / *ByDirection*.
Request Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *NationalLanguage.*
	Include-Translations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

7.4.2 StopTimetableRequest — Example

The following is an example of a ***StopTimetableRequest***.

```
<ServiceRequest>
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <StopTimetableRequest version="1.0">
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--=======TOPIC ==================================== -->
        <DepartureWindow>
            <StartTime>2004-12-17T09:30:47-05:00</StartTime>
            <EndTime>2004-12-17T10:30:47-05:00</EndTime>
        </DepartureWindow>
        <MonitoringRef>EH00001</MonitoringRef>
    </StopTimetableRequest>
</ServiceRequest>
```

7.5 StopTimetableSubscriptionRequest

7.5.1 StopTimetableSubscriptionRequest — Element

The ***StopTimetableSubscription*** requests the asynchronous delivery of the information described by a ***StopTimetableRequest***. The ***StopTimetableSubscription*** Policy parameters control the processing of the subscription.

Table 27 — StopTimetableSubscription — Attributes

StopTimetableSubscription			*+Structure*	Request for a subscription to the *Stop Timetable* Service.
Identity	***SubscriberRef***	0:1	*→ParticipantCode*	See SIRI Part 2 Common ***SubscriptionRequest*** parameters.
	SubscriptionIdentifier	**1:1**	*SubscriptionQualifier*	
Lease	***InitialTerminationTime***	**1:1**	*xsd:dateTime*	
Request	***StopTimetableRequest***	**1:1**	*+Structure*	See ***StopTimetableRequest***.
	Extensions	0:1	xsd:any*	Placeholder for user extensions.

7.5.2 StopTimetableSubscriptionRequest — Example

The following is an example of a ***StopTimetableSubscriptionRequest***:

```
<SubscriptionRequest>
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <StopTimetableSubscriptionRequest>
        <SubscriptionIdentifier>000234</SubscriptionIdentifier>
        <InitialTerminationTime>2004-12-17T15:30:47-05:01</InitialTerminationTime>
        <StopTimetableRequest version="0.1">
            <RequestTimestamp>2004-12-17T09:30:47-05:05</RequestTimestamp>
            <MonitoringRef>POIT5678</MonitoringRef>
        </StopTimetableRequest>
    </StopTimetableSubscriptionRequest>
</SubscriptionRequest>
```

7.6 StopTimetableDelivery

7.6.1 Introduction

The ***StopTimetableDelivery*** returns timetabled Stop Visits for a stop, with scheduled and times for each arrival or departure from the stop. It may be sent by the Producer to the Consumer as the result of either a direct ***StopTimetableRequest*** or an asynchronous ***StopTimetableSubscription***.

7.6.2 ServiceDelivery with a StopTimetableDelivery

One or more ***StopTimetableDelivery*** elements may be returned as part of a SIRI ***ServiceDelivery***.

Table 28 — ServiceDelivery / StopTimetableDelivery — Attributes

ServiceDelivery			+Structure	See SIRI Part 2 ***ServiceDelivery***
HEADER	***:::***	**1:1**	See ServiceDelivery	
Payload	***StopTimetableDelivery***	0:*	+Structure	See ***StopTimetableDelivery*** element.

7.6.3 StopTimetableDelivery — Element

A ***StopTimetableDelivery*** is made up of ***TimetabledStopVisit*** instances, each representing a CALL at the stop by a VEHICLE. (It may also contain ***TimetabledStopVisitCancellation*** elements.)

Each ***TimetabledStopVisit*** contains a set of subelements describing the VEHICLE's visit to the stop, including scheduled times; information about the journey may optionally be included in a ***VehicleJourneyInfo-Group.***

Table 29 — StopTimetableDelivery — Attributes

StopTimetableDelivery			*+Structure*	Delivery for *Stop Timetable* Service.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of *Stop Timetable* Service. Fixed.
LEADER	***:::***	**1:1**	xxx***Service-Delivery***	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***TimetabledStopVisit***	0:*	*+Structure*	A visit to a stop by a VEHICLE as an arrival and / or departure.
	TimetabledStopVisit-Cancellation	0:*	*+Structure*	A cancellation of visit to a stop by a VEHICLE as an arrival and / or departure.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

7.6.4 TimetabledStopVisit — Element

Each ***TimetabledStopVisit*** describes a CALL at a SCHEDULED STOP POINT by a VEHICLE making a timetabled VEHICLE JOURNEY, including scheduled passing times.

Table 30 — TimetabledStopVisit — Attributes

TimetabledStopVisit			*+Structure*	Timetabled visit of a VEHICLE to a SCHEDULED STOP POINT. May provide information about the arrival, the departure or both.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded by the AVMS.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used to refer to the Item later for server side cleardown.
	MonitoringRef	**1:1**	*→StopPoint-Code*	Reference to a monitoring point (LOGICAL DISPLAY) at which stop visits happen. May be a SCHEDULED STOP POINT or a timing point identifier. *DetailLevel*: minimum
Vehicle Journey	***TargetedVehicle-Journey***	**1:1**	*+Structure*	See ***TargetedVehicleJourney***.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

7.6.5 TargetedVehicleJourney — Element

Each ***TargetedVehicleJourney*** describes a VEHICLE JOURNEY timetable according to either the original or production timetable, including CALL details.

Table 31 — TargetedVehicleJourney — Attributes

TargetedVehicleJourney			*+Structure*	Targeted Vehicle Journey.
Vehicle Journey Identity	***LineRef***	**1:1**	*→LineCode*	Reference to a LINE.
	DirectionRef	**1:1**	*→DirectionCode*	Reference to the DIRECTION the VEHICLE is running along the JOURNEY PATTERN, for example, "in" or "out", “clockwise”. Distinct from a DESTINATION.
	FramedVehicle-JourneyRef	0:1	*+Structure*	A reference to the DATED VEHICLE JOURNEY that the VEHICLE is making. Unique with the data horizon of the service. See SIRI Part 2.
Journey-PatternInfo	***:::***	0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Vehicle Journey Info	***:::***	0:1	*Vehicle-JourneyInfoGroup*	See SIRI Part 2 ***VehicleJourneyInfoGroup***.
Operational Block	***:::***	0:1	*OperationalBlock-Group*	See SIRI Part 2 ***OperationalBlockGroup.***
Call	***TargetedCall***	0:1	*++Structure*	Information on targeted CALL. See ***TargetedCall***.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

7.6.6 TargetedCall — Element

Each ***TargetedCall*** describes a CALL at a monitoring reference point by a VEHICLE making a timetabled VEHICLE JOURNEY, including scheduled passing times.

Table 32 — TargetedCall — Attributes

TargetedCall			*+Structure*	Timetabled CALL of a VEHICLE to a Stop.
Identity	***StopPointRef***	0:1	*→StopPointCode*	Reference to a SCHEDULED STOP POINT. Defaults to that of context i.e. that specified on ***MonitoringRef***.
	VisitNumber	**1:1**	*VisitNumberType*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to a stop, the ***VisitNumber*** count is used to distinguish each separate visit.
	Order	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
Call Times	***TimingPoint***	0:1	*xsd:boolean*	Whether the stop is a timing point, i.e. times are measured at it. In Some systems this is a measure of data quality as non-timing points are interpolated.
	AimedArrivalTime	0:1	*xsd:dateTime*	Arrival time in either the original or Production Timetable.
	Arrival-PlatformName	0:1	*NLString*	Bay or platform name. Can be omitted if the same as the ***DeparturePlatformName***.
	ArrivalBoarding-Activity	0:1	*ArrivalBoarding-Activity Enumeration*	Type of alighting activity allowed at stop. *alighting* | *noAlighting* | *passthru*. Default is '*alighting'*.
	ArrivalOperatorRefs	0:*	*→OperatorCode*	OPERATORs of service up until arrival.. May change for departure. +SIRI v2.0.
	AimedDeparture-Time	0:1	*xsd:dateTime*	Departure time in either the original or Production Timetable.
	Departure-PlatformName	0:1	*NLString*	Bay or platform name.
	DepartureBoarding-Activity	0:1	*DepartureBoarding-Activity Enumeration*	Type of boarding allowed at stop. Default is '*boarding'*. *boarding* | *noBoarding*| *passthru*
	DepartureOperatorRefs	0:*	*→OperatorCode*	OPERATORs of service for departure and onwards. +SIRI v2.0.
	AimedLatest-Passenger-AccessTime	0:1	*xsd:dateTime*	Target latest time at which a PASSENGER should aim to arrive at the STOP PLACE containing the stop. This time may be earlier than the VEHICLE departure times and may include time for processes such as check-in, security, etc. (As specified by CHECK CONSTRAINT DELAYs in the underlying data) If absent assume to be the same as Earliest expected departure time, +SIRI v2.0
	AimedHeadway-Interval	0:1	*PositiveDuration*	For frequency based services, target interval of services at stop.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

7.6.7 TimetabledStopVisitCancellation — Element

A ***TimetabledStopVisitCancellation*** describes a cancellation of a previous ***TimetabledStopVisit***.

Table 33 — TimetabledStopVisitCancellation — Attributes

TimetabledStopVisitCancellation			*+Structure*	Cancellation of previous ***TimetabledStopVisit***.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded.
Identity	***ItemRef***	0:1	*ItemIdentifier*	Reference to a previous ***StopVisit*** which this item cancels.
StopVisit Reference	***MonitoringRef***	**1:1**	*→StopPoint-Code*	Reference to a monitoring point at which stop visits happen. May be a SCHEDULED STOP POINT or a display (LOGICAL DISPLAY) identifier.
	VisitNumber	0:1	*VisitNumber-Type*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to a stop, the VisitNumber is used to distinguish each separate visit.
Vehicle Journey Identity	***LineRef***	0:1	*→LineCode*	Identifier for the LINE.
	DirectionRef	0:1	*→DirectionCode*	Reference to the DIRECTION the VEHICLE is running along the JOURNEY PATTERN, for example, "in" or "out", “clockwise”. Distinct from a destination.
	FramedVehicle-JourneyRef	0:1	*+Structure*	A compound reference to the framed DATED VEHICLE JOURNEY that the VEHICLE is making. Unique with the data horizon of the service. See SIRI Part 2.
Journey-PatternInfo	**:::**	0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Message	***Reason***	0:*	*NLString*	Reason for cancellation. One per language (Unbounded 0:* since +SIRI v2.0).
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

7.6.8 StopTimetableDelivery — Example

The following is an example of a ***StopTimetableDelivery***.

```
<ServiceDelivery>
    <!--=======HEADER================================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <!--=======PAYLOAD================================================ -->
    <StopTimetableDelivery version="0.1d">
        <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
        <SubscriberRef>NADER</SubscriberRef>
        <SubscriptionRef>2004-12-17T09:30:47-05:00</SubscriptionRef>
        <Status>true</Status>
        <ValidUntil>2004-12-17T09:30:47-05:00</ValidUntil>
        <!--====FIRST ARRIVAL ============================================== -->
        <TimetabledStopVisit>
            <RecordedAtTime>2004-12-17T09:25:46-05:00</RecordedAtTime>
            <!-- IDENTITY GROUP -->
            <MonitoringRef>HLTST011</MonitoringRef>
            <TargetedVehicleJourney>
            <!-- JOURNEY IDENTITY GROUP -->
            <LineRef>Line123</LineRef>
            <DirectionRef>Out</DirectionRef>
            <FramedVehicleJourneyRef>
                <DataFrameRef>2004-12-17</DataFrameRef>
                <DatedVehicleJourneyRef>8765452</DatedVehicleJourneyRef>
            </FramedVehicleJourneyRef>
            <!-- JOURNEY PATTERN INFO GROUP -->
            <PublishedLineName>123</PublishedLineName>
            <!-- PASSING TIMES GROUP -->
```

```
                <TargetedCall>
                    <VisitNumber>1</VisitNumber>
                    <AimedArrivalTime>2004-12-17T09:40:46-05:00</AimedArrivalTime>
                    <AimedDepartureTime>2004-12-17T09:42:47-05:00</AimedDepartureTime>
                </TargetedCall>
            </TargetedVehicleJourney>
        </TimetabledStopVisit>
    </StopTimetableDelivery>
</ServiceDelivery>
```

8 Stop Monitoring Service [SM]

8.1 Purpose

The SIRI Stop Monitoring Service provides a stop-centric view of VEHICLE arrivals and departures at a designated stop. It can be used by displays and other presentation services to provide departure board and other presentations of timetable and real-time journey information both at stops and at a distance. The choice of data to display and the presentation is up to the client system. The service can be used in conjunction with the SIRI Stop Timetable service. For at stop use, the service can support cleardown identifiers to drive direct wireless cleardown of the displays signalled by the VEHICLE in proximity.

Each ***StopVisit****,* that is, visit to a stop by a VEHICLE, can include both scheduled and estimated arrival and departure times. The Stop Monitoring Service comprises the ***StopMonitoringRequest*** message to specify the information to be returned, and the ***StopMonitoringDelivery*** message, used to deliver the response payload. The ***StopMonitoringSubscription*** allows a subscriber to obtain asynchronous updates in the form of stop visits for a designated stop: the subscription contains an embedded ***StopMonitoringRequest****,* along with further parameters controlling the asynchronous delivery.

StopMonitoringRequest has topic parameters to filter the information by stop, LINE, and DIRECTION, etc., and policy parameters to control the amount of data returned. The ***StopMonitoringSubscription*** has further policy parameters, for example sensitivity threshold.

StopMonitoringMultipleRequest additionally allows the details for multiple stops to be requested at the same time.

The ***StopMonitoringDelivery*** returns information about one or more stop visits: associated with each ***MonitoredStopVisit*** is a ***MonitoredVehicleJourney*** instance, which may be populated to different levels of detail depending on the application requirements. The level of detail returned may range from very limited line and delay information suitable for a highly constrained display client communicating over limited bandwidth, to the data needed for a fully populated advanced station departure board, for example one that actively scrolls the journey details, including the full calling pattern for the journey.

The ‘vehicle’ tracked by a given stop visit may in fact be composite, that is part of a train. The visit of each individual Train Block part can be treated as a separate Stop Visit, allowing each part of the train to have its own accompanying ***MonitoredVehicleJourney*** information attached to describe the previous and onwards journeys of specific carriages. The parts can be related to the overall train using the ***TrainPartRef****.* The SIRI stop service may thus be used for both train, bus and train information.

8.2 Reference Data

8.2.1 Introduction

The Stop Monitoring Service requires the participants to have agreed data reference models for (i) Stop Point (i.e. SCHEDULED STOP POINT or Stop Place), (ii) LINE, and (iii) DIRECTION. The ***StopPoints-DiscoveryRequest*** can be used to obtain a list of stops with real-time support. The ***LinesDiscoveryRequest*** can be used to obtain a list of LINEs, DIRECTIONs and destinations allowed in requests.

8.2.2 Content Referencing

The ***ContentReferencing*** capability is an optional capability, allowing implementers to optimize the use of bandwidth if necessary. To identify elements identified by codes, such as stops and DIRECTIONs, systems may decide whether they wish to exchange just references, full names, or both. If only references are exchanged, then the client system shall hold names itself.

8.2.3 Direct Cleardown

Vehicles approaching the near vicinity of a bus stop may use a direct wireless signal to trigger the direct cleardown of their arrival notice from the display. To support direct cleardown an additional ***CleardownIdentifier*** may be associated with a Stop Visit; this may have an arbitrary short encoding suitable for constrained bandwidth of a local wireless signal. Both VEHICLE and AVMs shall be provisioned with the same identifier.

8.3 Capability and Permission Matrices

8.3.1 Capability Matrix

The following set of required and optional capabilities is defined for the Stop Monitoring service. If the service supports Capability Discovery the ***StopMonitoringCapabilitiesRequest*** / ***StopMonitoring-CapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 34 — StopMonitoringCapabilities Matrix

StopMonitoringCapabilities			*+Structure*	Capabilities describing implementation of Stop Monitoring service.
inherit	*:::*	**1:1**	*See xxxCapability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***	0:1	*+Structure*	Which optional filtering features are supported?
	DefaultPreview-Interval	**1:1**	*Positive-DurationType*	Default Preview Interval.
	ByStartTime	0:1	*xsd:boolean*	Whether a start time other than now can be specified for preview interval. Default is *'true'*.
	FilterByMonitoringRef	**1:1**	*xsd:boolean*	Whether results can be filtered by Monitoring point. Required Capability: Fixed as *true*.
	FilterByLineRef	**1:1**	*xsd:boolean*	Whether results can be filtered by ***LineRef***. Required Capability: Fixed as *true*.
	FilterByDirectionRef	0:1	*xsd:boolean*	Whether results can be filtered by ***DirectionRef***. Default is *'true'*.
	FilterByDestination	0:1	*xsd:boolean*	Whether results can be filtered by ***DestinationRef***. Default is *'false'*.
	FilterByVisitType	0:1	*xsd:boolean*	Whether results can be filtered by ***VisitType***, e.g. arrivals, departures. Default is *'true'*

Request Policy	***RequestPolicy***		0:1	*+Structure*	Which features of ***RequestPolicy*** are supported by service?
		Language	**1:***	*xsd:language*	National languages used by service.
		Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
				choice	One of the following Location formats:
	a	***GmlCoordinateFormat***	**1:1**	*SrsNameType*	Default coordinate format is given by a GML value.
	b	***WgsDecimalDegrees***		*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
		UseReferences	0:1	*xsd:boolean*	Whether results can return references for stops. Default is *'true'*.
		UseNames	0:1	*xsd:boolean*	Whether results can return names for stop. Default is *'true'*.
		HasDetailLevel	0:1	*xsd:boolean*	Whether Detail level filtering is supported. Default *false*.
		DefaultDetailLevel	0:1	*StopMonitoring-RequestDetailLevel Enum*	Default Detail level if none specified on request. *minimum* | *basic* | *normal* | *calls* | *full* Default Normal.
		HasMaximumVisits	0:1	*xsd:boolean*	Whether results can be limited to a maximum number. Default is *'true'*.
		HasMinimumVisits-PerLine	0:1	*xsd:boolean*	Whether results can be limited to include a minimum number per LINE. Default is *'true'*.
		HasMinimumVisits-PerVia	0:1	*xsd:boolean*	Whether results can be limited to include a minimum number per VIA (i.e. JOURNEY PATTERN). +SIRI v2.0. Default is *'false'*.
		HasNumberOf-OnwardsCalls	0:1	*xsd:boolean*	If system can return detailed calling pattern, whether a number of ONWARDS CALLs to include can be specified. Default is *false*
		HasNumberOf-PreviousCalls	0:1	*xsd:boolean*	If system can return detailed calling pattern, whether a number of PREVIOUS CALLs to include can be specified. Default is *'false'*.
Subscript-ion-Policy	***SubscriptionPolicy***		0:1	*+Structure*	Which features of Subscription Policy are supported by service?
		HasIncremental-Updates	0:1	*xsd:boolean*	Whether incremental updates can be specified for updates. Default is *'true'*.
		HasChangeSensitivity	0:1	*xsd:boolean*	Whether change threshold can be specified for updates. Default is *'true'*.

Access Control	***AccessControl***	0:1	*+Structure*	Which optional Access Control features are supported by service?
	RequestChecking	**1:1**	*xsd:boolean*	Whether Access Control of requests is supported. Default is *'false'*.
	CheckOperatorRef	0:1	*xsd:boolean*	If access control is supported, whether access control by OPERATOR is supported. Default is *'true'*.
	CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
	CheckMonitoringRef	0:1	*xsd:boolean*	If access control is supported, whether access control by monitoring point (LOGICAL DISPLAY) is supported. Default is *'true'*.
Response	***ResponseFeatures***	0:1	*+Structure*	Which features of Response data are supported by service?
	HasLineNotice	0:1	*xsd:boolean*	Whether service supports Line Notices. Default is *'true'*.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.3.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***StopMonitoringCapabilitiesResponse*** response can include the access permissions for the requestor participant to access data.

Table 35 — Stop Monitoring Service — Permissions

StopMonitoringPermissions			*+Structure*	Permissions to use implementation of *Stop Monitoring* service.
Inherit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant.
	StopMonitorPermissions	0:1	*+Structure*	Monitor ***(***LOGICAL DISPLAY***)*** permissions for participant.

8.4 StopMonitoringRequest

8.4.1 StopMonitoringRequest — Element

The ***StopMonitoringRequest*** can be used in both a direct request, and for a subscription. If used for a subscription, additional ***StopMonitoringSubscriptionPolicy*** parameters can be specified.

Only a single stop or display point may be specified in a single ***StopMonitoringSubscription***; a ***SubscriptionRequest*** may contain multiple service subscriptions per request, one for each stop. In this case, a single ***ServiceDelivery*** will be returned with separate ***StopMonitoringDelivery*** messages for each SCHEDULED STOP POINT.

The primary ***Topic*** term on the request is the SCHEDULED STOP POINT for which information is to be returned. Additional topic filtering on LINE and DIRECTION is also allowed. Other properties such as OPERATOR may also be used. Filtering is additive (though some parameters are mutually exclusive).

Table 36 — StopMonitoringRequest — Attributes

StopMonitoringRequest			*+Structure*	Request for information about arrival/departure visits at a stop.
Attrib-utes	***Version***	**1:1**	*VersionString*	Version identifier of Stop Monitoring Service, e.g. '1.0c'.
End-point Proper-ties	***Request-Timestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	Message-Identifier	0:1	*MessageQualifier*	
Topic	***Preview-Interval***	0:1	*PositiveDuration-Type*	Forward duration for which Stop Visits should be included, that is, interval before predicted arrival at the stop for which to include visits: only journeys which will arrive or depart within this time span will be returned.
	StartTime	0:1	*xsd:dateTime*	Initial start time for ***PreviewInterval***. If absent, then current time is assumed. Must be within data Horizon.
	Monitoring-Ref	**1:1**	*→Monitoring¬Code*	Reference to a Stop Monitoring point for which Stop Visits will be returned May be an identifier of a SCHEDULED STOP POINT or a display. (LOGICAL DISPLAY)
	LineRef	0:1	*→LineCode*	Filter the results to include only Stop Visits for VEHICLEs for the given LINE.
	DirectionRef	0:1	*→DirectionCode*	Filter the results to include only Stop Visits for VEHICLEs running in a specific relative DIRECTION, for example, "inbound" or "outbound". (Direction does not specify a destination.) Optional SIRI capability: *TopicFiltering / ByDirection.*
	Destination-Ref	0:1	*→StopPointCode*	Filter the results to include only visits for VEHICLEs going to the specified DESTINATION. Optional SIRI capability: *TopicFiltering* / *ByDestination.*
	OperatorRef	0:1	*→OperatorCode*	Filter the results to include only Stop Visits for VEHICLEs run by the specified OPERATOR. Optional SIRI capability: *TopicFiltering / ByOperator.*
	StopVisit-Types	0:1	*StopVisitTypeEnum*	Whether to include arrival visits, departure visits, or both. Default is '*all*'. *all* | *departures* | *arrivals* Optional SIRI capability: *TopicFiltering* / *ByVisitType.*

Request Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *NationalLanguage.*
	Include-Translations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
	Maximum-StopVisits	0:1	*xsd:positive-Integer*	The maximum number of arrival or departure visits to include in a given delivery. The first *n* Stop Visits within the look-ahead window are included. Only visits within the ***PreviewInterval*** are returned. The ***MinimumStopVisitsPerLine*** parameter can be used to reduce the number of entries for each LINE within the total returned.
			choice	
	a ***Minimum-Stop-Visits-PerLine***	0:1	*xsd:nonNegative-Integer*	The requested minimum number of arrival or departure visits for a given LINE to include in a given delivery. If there are more visits within the ***PreviewInterval*** than allowed by ***MaximumStopVisits***, and a ***MinimumStopVisitsPerLine*** value is specified, then at least the minimum number will be delivered for each LINE. That is, the first available visits for each LINE within the ***PreviewInterval*** will be included, even if the Stop Visits are later than those for some other LINE for which the minimum number of Stop Visits has already been supplied. This allows the Consumer to obtain at least one entry for every available LINE with VEHICLEs approaching the stop. Only visits within the ***PreviewInterval*** are returned.
	b ***Minimum Stop-Visits-PerLine-Via***	0:1	*xsd:nonNegative-Integer*	The minimum number of Stop Visits for a given LINE and VIA combination to include in a given delivery. As for MinimumStopVisitsPerLine but with VIA also taken into account. SIRI+v2.0
	Maximum-TextLength	0:1	*xsd:positive-Integer*	Maximum length of text to return for text elements. Applies to names, messages and other text elements. If unspecified no limit.
	StopVisit-DetailLevel	0:1	*StopVisitDetail-LevelEnum*	Level of detail to include in response. Default is '*normal'*. Optional SIRI capability: ***DetailLevel*** (if absent, must support *Normal). minimum* | *basic* | *normal* | *calls* | *full*
	Include-Situations	0:1	*xsd:boolean*	Whether any related SITUATIONs should be included in the ServiceDelivery. Default is 'false'. +SIRI v2.0
	Maximum-NumberOf-Calls	0:1	*+Structure*	If CALLs are to be returned, maximum number of CALLs to include in response. If absent, include all CALLs. Optional SIRI capability: ***StopVisitCalls***.
	Previous	0:1	*xsd:nonNegative-Integer*	Maximum number of PREVIOUS CALLs to include. Zero for *DetailLevel*:.none. Only applies if ***MaximumNumber-OfCalls*** specified.. If ***MaximumNumberOfCalls*** specified but ***MaximumNumberOfCalls***.previous absent, include all PREVIOUS CALLs.
	Onwards	0:1	*xsd:nonNegative-Integer*	Maximum number of ONWARDS CALLs to include. Zero for *DetailLevel*:.none. Only applies if MaximumNumberOfCalls specified. If ***MaximumNumberOf-Calls*** specified but ***MaximumNumberOfCallsOnwards*** absent, include all ONWARDS CALLs.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.4.2 Use of Preview Interval

8.4.2.1 General

The preview interval defines the time window before the actual arrival of the VEHICLEs in the SCHEDULED STOP POINT within which any schedule variance information should be transmitted. If the VEHICLE is not en-route at this point, the information transmitted relates to the planning data.

8.4.2.2 *StopVisitType* — Allowed values

Allowed values for ***StopVisitType*** (*StopVisitTypeEnumeration*).

Table 37 — *StopVisitType* —Allowed Values

Value	Description
all	Include both arrivals & departures.
departures	Include only departures.
arrivals	Include only arrivals.

8.4.3 Use of Maximum and Minimum Number of Trips

8.4.3.1 General

The ***PreviewInterval***, ***MaximumStopVisits*** and ***MinimumStopVisitsPerLine*** elements can be used to optimize data traffic and to ensure that the set of Stop Visits returned is as useful as possible to passengers. This is useful in partiucalr where the majority of joruneys are to one destination but it is helpful tio include journeys to other destinations as well.

— The ***PreviewInterval*** and ***MaximumStopVisits*** limit the number of ***StopVisit*** instances returned.

— The ***MinimumStopVisitsPerLine*** parameter can be used to ensure that if only a limited number of visits are returned, then that at least one ***StopVisit*** is included for each available LINE.

— The ***MinimumStopVisitsPerLineVia*** parameter can be used to ensure that if only a limited number of visits are returned, then that at least one ***StopVisit*** is included for each available LINE and VIA combination. (+SIRI v2.0).

The following Table 38 shows examples of various combinations of stop filtering.

Table 38 — Examples of StopVisit Filtering

		Example 1	Example 2	Example 3	Example 4	Example 5
	Current Time	10:15	10:15	10:15	11:12	10:15
Request	***PreviewInterval***	40	40	60	60	60
	MaxStopVisits	--	8	8	6	10
	MinimumStopVisits-PerLine	--	--	2	1	--
	MinimumStopVisits-PerLineVia	--	1	--	--	--
	Line	--	--	--	--	A
	MonitoredStopVisits					
1	123 Line A 11:10	123 A 11:10	123 A 11:10	123 A 11:10	--	123 A 11:10
2	125 Line A: 11:12	125 A:11:12	125 A 11:12	125 A 11:12	125 A 11:12	125 A 11:12
3	226 Line C 11:18	226 C 11:18	226 C 11:18	226 C 11:18	226 C 11:18	--
4	512 Line B 11:25 via X	512 B 11:25	512 B 11:25 viaX	512 B 11:25	512 B 11:25	--
5	514 Line B 11:27 via X	514 B 11:27		514 B 11:27	514 B 11:27	--
6	515 Line B 11:30 via Y	515 B 11:30	515 B 11:30 viaY	--	515 B 11:30	--
7	227 Line C 11:31 via P	227 C 11:31	227 C 11:31 viaP	227 C 11:31	--	--
8	228 Line C 11:34 viaQ	228 C 11:34	228 C 11:34 viaQ	--	--	--
9	127 Line D 11:37	127 D 11:37	--	127 D 11:37	127 D 11:37	--
10	128 Line A 12:01	--	--	128 A 12:01	--	128 A 12:01

8.4.3.2 StopMonitoringRequestDetailLevel — Allowed values

Different consumers will have different requirements for data. The ***StopVisitDetailLevel*** parameter on the request can be used to control how much detail is returned for each Stop Visit (see Table 39 below). The Normal level is intended to be a useful standard level of detail.

Table 39 — StopMonitoringRequestDetailLevels — Allowed values

Detail Level	Definition.
minimum	Return only minimum data.
basic	Return useful basic minimum data.
normal	Return additional information about stop and destination.
calls	Return additional information including previous or onward calling pattern, modulated by the ***NumberOfCalls Onwards*** and ***Previous*** elements.
full	Return all information including full calling pattern.

8.5 StopMonitoringMultipleRequest

8.5.1 StopMonitoringMultipleRequest — Element

The ***StopMonitoringMultipleRequest*** can be used in both a direct request, and for a subscription. If used for a subscription, additional ***StopMonitoringSubscriptionPolicy*** parameters can be specified.

It allows the requesting of multiple stops at the same time.

Table 40 — StopMonitoringMultipleRequest — Attributes

StopMonitoringMultiple-Request			*+Structure*	Request for information about arrival/departure visits at several stops.
Attrib-utes	***Version***	**1:1**	*VersionString*	Version identifier of Stop Monitoring Service, e.g. '1.0c'.
End-point Proper-ties	***Request-Timestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	Message-Identifier	0:1	*MessageQualifier*	
Filter	***Stop-Monitoring-Filter***	0:1	*+Structure*	Filter for StopMonitoringPoints for which data is to be returned.
	Request-Topics	0:1	*group*	As for ***StopMonitoringRequest***
	Request-Policies	0:1	*group*	As for ***StopMonitoringRequest***
	Extensions	0:1	xsd:any*	Placeholder for user extensions.

8.5.2 StopMonitoringRequest — Example

The following is an example of a *StopMonitoringRequest*.

```
<ServiceRequest>
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <StopMonitoringRequest version="1.0">
    <!-- All LINE77services from stop EH00001to destination PLACE457 in the next 30 mins-->
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--========TOPIC ================================== -->
        <PreviewInterval>P30M</PreviewInterval>
        <MonitoringRef>EH00001</MonitoringRef>
        <LineRef>LINE77</LineRef>
        <!--========POLICY=========================================-->
        <MaximumStopVisits>7</MaximumStopVisits>
        <MinimumStopVisitsPerLine>2</MinimumStopVisitsPerLine>
        <MaximumTextLength>20</MaximumTextLength>
        <StopMonitoringDetailLevel>normal</StopMonitoringDetailLevel>
    </StopMonitoringRequest>
</ServiceRequest>
```

8.6 StopMonitoringSubscriptionRequest

8.6.1 StopMonitoringSubscriptionRequest — Element

The ***StopMonitoringSubscription*** requests the asynchronous delivery of the information described by a ***StopMonitoringRequest***. The ***StopMonitoringSubscription*** Policy parameters control the processing of the subscription.

Table 41 — StopMonitoringSubscription — Attributes

StopMonitoringSubscription			*+Structure*	Request for a subscription to the SIRI ***StopMonitoring*** Service
Identity	***SubscriberRef***	0:1	*→Participant-Code*	See SIRI Part 2 Common ***SubscriptionRequest*** parameters.
	Subscription-Identifier	**1:1**	*Subscription-Qualifier*	
Lease	***Initial-Termination-Time***	**1:1**	*xsd:dateTime*	
Request	***StopMonitoring Request***	**1:1**	*+Structure*	See ***StopMonitoringRequest***.
Policy	***Incremental-Updates***	0:1	*xsd:boolean*	Whether the producer should only provide updates to the last data returned, i.e. additions, modifications and cancellations, or always return the complete set of current data, Default is *'true'*, i.e. once the initial transmission has been made, return only incremental updates. If *false,* each subscription response will contain the full information as specified in this request. Optional SIRI capability: ***IncrementalUpdates***.
	ChangeBefore-Updates	0:1	*Positive-DurationType*	The amount of change to the arrival or departure time that can happen before an update is sent (i.e. if *ChangeBeforeUpdate* is set to 2 min, the subscriber will not be told that a VEHICLE is 30 s delayed – an update will only be sent when the VEHICLE is at least 2 min delayed). Optional SIRI capability: ***ChangeSensitivityThreshold***.
	Extensions	0:1	xsd:any*	Placeholder for user extensions.

8.6.2 Sensitivity Threshold

When a subscription is first made the current Stop Visits are transmitted, thereafter, in order to avoid constant updates of insignificant changes, only changes are sent according to the specified ***ChangeBeforeUpdates*** parameter. Data is deemed to have changed if at least one of the following pieces of information has changed:

— *Variation from schedule adherence:* the predictions for the stop have changed by more than the sensitivity threshold.

— *Change of Platform:* a change of arrival or departure platform is material to the passenger and should be transmitted.

— *Completion of journey*: a change remains active until the data assigned to the subscription is explicitly retrieved.

The sensitivity threshold value can be assigned within the subscription. It defines the time span, after which the system regards the change in schedule deviation significant enough to demand communication. With a

sensitivity of 2 min, for example, changes in deviations of 2, 4, 6 min, etc. are reported. The change is with respect to the last transmitted arrival and/or departure time.

8.6.3 StopMonitoringSubscriptionRequest — Example

The following is an example of a ***StopMonitoringSubscriptionRequest***:

```
<SubscriptionRequest>
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <!-- Next 10 services from stop POIT5678 to destination PLACE457 in the next 30 mins -->
        <StopMonitoringSubscriptionRequest>
            <SubscriptionIdentifier>000234</SubscriptionIdentifier>
            <InitialTerminationTime>2004-12-17T15:30:47-05:01</InitialTerminationTime>
            <StopMonitoringRequest version="1.0">
                <RequestTimestamp>2004-12-17T09:30:47-05:05</RequestTimestamp>
                <PreviewInterval>PT30M</PreviewInterval>
                <MonitoringRef>POIT5678</MonitoringRef>
                <MaximumStopVisits>10</MaximumStopVisits>
            </StopMonitoringRequest>
            <IncrementalUpdates>true</IncrementalUpdates>
            <ChangeBeforeUpdates>PT2M</ChangeBeforeUpdates>
        </StopMonitoringSubscriptionRequest>
        <!-- All Line23 services from stop POIT5678 to destination PLACE457 in the next 30 mins-->
        <StopMonitoringSubscriptionRequest>
            <SubscriptionIdentifier>000235</SubscriptionIdentifier>
            <InitialTerminationTime>2004-12-17T15:30:47-05:01</InitialTerminationTime>
            <StopMonitoringRequest version="1.0">
                <RequestTimestamp>2004-12-17T09:30:47-05:06</RequestTimestamp>
                <PreviewInterval>PT30M</PreviewInterval>
                <MonitoringRef>POIT5678</MonitoringRef>
                <LineRef>LINE23</LineRef>
                <DestinationRef>PLACE457</DestinationRef>
            </StopMonitoringRequest>
            <IncrementalUpdates>true</IncrementalUpdates>
            <ChangeBeforeUpdates>PT2M</ChangeBeforeUpdates>
        </StopMonitoringSubscriptionRequest>
</SubscriptionRequest>
```

8.7 StopMonitoringDelivery

8.7.1 Introduction

The ***StopMonitoringDelivery*** returns Stop Visits for a stop, with scheduled and predicted times for each arrival or departure from the stop.

It may be sent by the Producer to the Consumer as the result of either a direct ***StopMonitoringRequest***, or an asynchronous ***StopMonitoringSubscription***.

8.7.2 ServiceDelivery with a StopMonitoringDelivery — Element

One or more ***StopMonitoringDelivery*** elements may be returned as part of a SIRI ***ServiceDelivery***.

Table 42 — ServiceDelivery / StopMonitoringDelivery — Attributes

ServiceDelivery			*+Structure*	See SIRI Part 2 ***ServiceDelivery***
HEADER	*:::*	**1:1**	See ServiceDelivery	
Payload	***StopMonitoringDelivery***	**1:***	*+Structure*	See *StopMonitoringDelivery* element.

8.7.3 StopMonitoringDelivery — Element

A ***StopMonitoringDelivery*** is made up of ***MonitoredStopVisit*** instances, each representing a CALL at the SCHEDULED STOP POINT by a VEHICLE, and/or ***MonitoredStopVisitCancellation*** instances to remove ***MonitoredStopVisit*** instances.

If the Vehicle Journey is a Train made up of several parts that may merge or join, there should be a separate ***MonitoredStopVisit*** for each train part.

Each ***MonitoredStopVisit*** contains a set of subelements describing the VEHICLE's visit to the stop, including scheduled and/or real-time times, and also a ***MonitoredVehicleJourney*** element, which describes the VEHICLE's route and associations with the ***DatedVehicleJourney*** that it follows, and other entities. The journey description may include detailed information about origin & destination place names and via points. It may also optionally include detailed information of individual CALLs, that is, stops in stop sequence of the journey, with target, and estimated times. Detailed calling information will not normally be used for bus services, but may be of use for train display boards. The level of detail included for each ***MonitoredStopVisit*** element may vary by implementation and by request. Common data elements are used to structure the response, which may be populated as appropriate to the implementation capabilities and requirements to one of four specified levels of detail, as specified by a ***StopMonitoringDetailLevel***.

The ***Note*** element allows one or more arbitrary text strings to be associated with the whole stop, a VEHICLE JOURNEY or an individual JOURNEY PATTERN.

Table 43 — StopMonitoringDelivery — Attributes

StopMonitoringDelivery			*+Structure*	Delivery for Stop Monitoring Service.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Stop Monitoring Service. Fixed.
LEADER	***:::***	:::	xxx***Service-Delivery***	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***MonitoringRef***	0:*	*→Monitoring¬Code*	Reference to a stop monitoring point (LOGICAL DISPLAY) that was requested. This can be used to return the requested Monitoring Point even if there are no active stop visits for the stop. SIRI v1.3 The cardinality is 0:* since in case of subscription information on multiple MonitoringRef can be delivered in a single SIRI message.
	Monitored-StopVisit	0:*	*+Structure*	A visit to a SCHEDULED STOP POINT by a VEHICLE as an arrival and /or departure. *DetailLevel*: minimum.
	Monitored-StopVisit-Cancellation	0:*	*+Structure*	Reference to a previously communicated which should now be removed from the arrival/departure board for the stop. *DetailLevel*: minimum.
	StopLineNotice	0:*	*+Structure*	A notice concerning a LINE. *DetailLevel*: minimum.
	StopLineNotice-Cancellation	0:*	*+Structure*	Reference to an previously communicated ***StopLineNotice*** which should now be removed from the arrival/departure board for the stop. *DetailLevel*: minimum.
	ServiceException	0:*	*+Structure*	Information about why data is unavailable for the functional service. (+SIRI v2.0) *DetailLevel*: basic.
	Note	0:*	*NLString*	Message associated with delivery. *DetailLevel*: basic.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.4 MonitoredStopVisit — Element

8.7.4.1 General

Each ***MonitoredStopVisit*** describes a CALL at a SCHEDULED STOP POINT by a VEHICLE making a monitored VEHICLE JOURNEY, including scheduled and/or predicted real-time times.

Each ***MonitoredStopVisit*** has an identifier: this can be used to reference trips when sending incremental updates – for example the ***MonitoredStopVisitCancellation*** indicates a previous element has been removed.

Table 44 — MonitoredStopVisit — Attributes

MonitoredStopVisit			*+Structure*	Visit of a VEHICLE to a stop. May provide information about the arrival, the departure or both.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded. *DetailLevel*: minimum.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used to refer to the item later for server side cleardown. *DetailLevel*: minimum.
Currency	***ValidUntilTime***	0:1	*xsd:dateTime*	Time until which data is valid. (+SIRI v2.0). This allows an override at the individual VISIT level of the value at the delivery level.
StopVisit-Reference	***MonitoringRef***	**1:1**	*→Monitoring¬Code*	Reference to a Stop Monitoring point at which visits happen. May be a SCHEDULED STOP POINT or a display identifier. *DetailLevel*: minimum.
	Monitoring-Name	0:*	*NLString*	Name to use to describe monitoring point (SCHEDULED STOP POINT or (LOGICAL DISPLAY)). Normally Consumer will already have access to this in its reference data, but may be included to increase usability of SIRI LITE services (+SIRI v2.0). One per language
	ClearDownRef	0:1	*→ClearDownCode*	Identifier associated with ***MonitoredStopVisit*** for use in direct wireless communication between VEHICLE and stop display. Cleardown codes are short arbitrary identifiers suitable for radio transmission. Their scope may be transient, that is, they may be unique only to a day and sector. *DetailLevel*: normal.
JourneyInfo	***Monitored-VehicleJourney***	**1:1**	*MonitoredVehicle-JourneyStructure*	Provides real-time information about the monitored VEHICLE JOURNEY which this VEHICLE is running. *DetailLevel*: minimum.
Message	***StopVisitNote***	0:*	*NLString*	Message associated with delivery. *DetailLevel*: basic. There may be multiple notes and translations into different languages.
Facility	***StopFacility***	0:1	*→FacilityCode*	Facility associated with stop visit. SIRI 1.3
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

There are various calling times at the stop that may be returned for a ***MonitoredStopVisit***. The target times are the timetabled times: these will be according to the latest timetable available. If monitoring is not available there will be only target times available. At least one time or frequency shall be returned.

8.7.4.2 MonitoredStopVisit / MonitoredVehicleJourney — Element

Each ***MonitoredStopVisit*** has a ***MonitoredVehicleJourney*** element, which associates the ***MonitoredStop-Visit*** with the ***MonitoredVehicleJourney*** being made by the Vehicle arriving at the stop. The ***Monitored-VehicleJourney*** may be populated to different levels of detail.

Table 45 — MonitoredStopVisit / MonitoredVehicleJourney — Attributes

MonitoredVehicleJourney			*+Structure*	Provides real-time information about the VEHICLE JOURNEY along which a VEHICLE is running.
Vehicle Journey Identity	***LineRef***	0:1	*→LineCode*	Reference to a LINE. *DetailLevel*: minimum.
	DirectionRef	0:1	*→DirectionCode*	Reference to a DIRECTION the VEHICLE is running along the LINE, for example, "in" or "out", “clockwise”. Distinct from a DESTINATION. *DetailLevel*: minimum.
	FramedVehicle-JourneyRef	0:1	*+FramedVehicle-JourneyRef-Structure*	A reference to the DATED VEHICLE JOURNEY that the VEHICLE is making. Unique with the data horizon of the service. See SIRI Part 2. *DetailLevel*: basic.
Journey-Pattern-Info	***:::***	0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***. *DetailLevel*: normal.
Vehicle-Journey-Info	***:::***	0:1	*Vehicle-JourneyInfoGroup*	See SIRI Part 2 ***VehicleJourneyInfoGroup***. *DetailLevel*: normal.
DisruptionGroup	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***DisruptionGroup***. *DetailLevel*: normal.
Journey-ProgressInfo	***:::***	0:1	*JourneyProgress-Group*	See SIRI Part 2 ***JourneyProgressGroup***. *DetailLevel*: normal.
Train-Operational Info	***TrainBlockPart***	0:*	*TrainBlockPart-Structure*	Associates Stop Visit with a part of a TRAIN: for use when trains split or merge. *DetailLevel*: normal.
	NumberOf-BlockParts	**1:1**	*xsd:positiveInteger*	Total number of block parts making up the TRAIN of which this is part.
	TrainPartRef	**1:1**	*→TrainPartCode*	Reference to a TRAIN BLOCK PART.
	PositionOf TrainBlockPart	0:*	*NLString*	Description of position of ***TrainBlockPart*** within Train to guide passengers where to find it. E.g. 'Front four coaches'. One per language (Unbounded 0:* since +SIRI v2.0).
OperationalInfo	***:::***	0:1	*OperationalInfo-Group*	See SIRI Part 2 ***OperationalInfoGroup***. ***BlockRef*** & ***CourseOfJourneyRef*** *(from **OperationnalBlockGroup**): DetailLevel*: normal. ***VehicleRef:*** *DetailLevel*: basic.

Train Operati-onal Info	***TrainNumbers***	0:*	*sequence*	One or more. +SIRI v2.0
	TrainNumber-Ref	**1:1**	*→TrainNumber*	UE regulation 454/2011 primary code or UIC Train Number. +SIRI v2.0
	JourneyParts	0:*	*sequence*	One or more. +SIRI v2.0
	JourneyPart-Info	**1:1**	*+Structure*	Information about Parts of JOURNEY +SIRI v2.0
	Journey-PartRef	**1:1**	*→JourneyPart-Code*	Reference to a JOURNEY PART. +SIRI v2.0
	Train-NumberRef	0:1	*→TrainNumber*	Reference to TRAIN NUMBER for a JOURNEY PART. +SIRI v2.0
	Operator-Ref	0:1	*→OperatorCode*	Reference to OPERATOR of a JOURNEY PART. +SIRI v2.0
Calling Pattern	***PreviousCalls***	0:1	*+Structure*	Information on stops called at previously, the origin stop and all intermediate stops up to but not including the current stop. Should only be included if the detail level was requested. *DetailLevel*: calls.
	PreviousCall	**1:***	*+Structure*	Information on a stop called at previously. See ***PreviousCall*** element.
	MonitoredCall	0:1	*+Structure*	Information about a CALL at a SCHEDULED STOP POINT. See ***MonitoredCall*** element. *DetailLevel*: minimum.
	OnwardCalls	0:1	*+Structure*	Information on CALLs at the intermediate stops beyond the current stop, up to and including the DESTINATION. Should only be included if the detail level was requested. *DetailLevel*: calls.
	OnwardCall	**1:***	*+Structure*	Information on an onward stop CALL. See ***OnwardCall*** element.
	IsCompleteStop-Sequence	0:1	*xsd:boolean*	Whether the CALL sequence is simple, i.e. represents every CALL of the ROUTE and so can be used to replace a previous CALL sequence. Default is *'false'*. *DetailLevel*: calls.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.4.3 MonitoredVehicleJourney / PreviousCall — Element

The ***PreviousCall*** element describes a CALL which has already been made in the ***MonitoredVehicle-Journey***.

It will only be populated for a ***MonitoredStopVisit*** if a ***StopMonitoringDetailLevel*** of full was requested. The number of onward CALLs to include is given by the ***MaximumNumberOfCalls*** - previous element.

Table 46 — MonitoredStopVisit / MonitoredVehicleJourney / PreviousCall — Attributes

PreviousCall		0:*	*+Structure*	Information on a stop previously called at by the VEHICLE before the current stop.
Stop Identity	***StopPointRef***	0:1	*→StopPointRef*	Reference to a SCHEDULED STOP POINT.
	VisitNumber	0:1	*VisitNumberType*	Sequence of visit to a stop within a JOURNEY PATTERN.
	Order	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Name of SCHEDULED STOP POINT. One per language (Unbounded 0:* since +SIRI v2.0).
Real-time	***VehicleAtStop***	0:1	*xsd:boolean*	Whether VEHICLE is at stop at the current time.
Arrival Times	***AimedArrivalTime***	0:1	*xsd:dateTime*	Arrival time of VEHICLE in either the original or Production Timetable.
	ActualArrivalTime	0:1	*xsd:dateTime*	Observed time of arrival of VEHICLE.
	ExpectedArrivalTime	0:1	*xsd:dateTime*	Estimated time of arrival of VEHICLE.
Depart-ure *Times*	***AimedDepartureTime***	0:1	*xsd:dateTime*	Departure time of VEHICLE in either the original or Production Timetable.
	ActualDepartureTime	0:1	*xsd:dateTime*	Observed time of departure of VEHICLE.
	ExpectedDeparture-Time	0:1	*xsd:dateTime*	Estimated time of departure of VEHICLE.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.4.4 MonitoredVehicleJourney / MonitoredCall — Element

8.7.4.4.1 General

The ***MonitoredCall*** element describes the current CALL (or if the vehicle is between stops the most recent CALL) made at a SCHEDULED STOP POINT of the ***MonitoredVehicleJourney***.

For a ***StopMonitoringRequest*** the ***MonitoredCall*** is the stop requested in the request ***MonitoringRef*** *(LOGICAL DISPLAY)*.

For a ***VehicleMonitoringRequest*** the ***MonitoredCall*** is the stop at which the VEHICLE is or has most recently left.

Table 47 — MonitoredStopVisit / MonitoredVehicleJourney / MonitoredCall — Attributes

MonitoredCall			*+Structure*	Information about a CALL at the current or most recent stop.
Stop Identity	***StopPointRef***	0:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT. Defaults to that of context i.e. that specified on ***MonitoredStopVisit***. *DetailLevel*: minimum
	VisitNumber	0:1	*VisitNumber-Type*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to the same stop, the ***VisitNumber*** is used to distinguish each separate visit. *DetailLevel*: minimum.
	Order	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful. *DetailLevel*: minimum
	StopPointName	0:*	*NLString*	Name of SCHEDULED STOP POINT. One per language (Unbounded 0:* since +SIRI v2.0).
Call Real-time	***VehicleAtStop***	0:1	*xsd:boolean*	Whether VEHICLE is at stop at the current time. If absent, unknown *DetailLevel*: normal.
	Vehicle-LocationAtStop	0:1	*Location-Structure*	Location that VEHICLE will take up at SCHEDULED STOP POINT. *DetailLevel*: normal.
Call Rail	***ReversesAtStop***	0:1	*xsd:boolean*	Whether VEHICLE reverses at stop. Default is *'false'*. *DetailLevel*: normal.
	Platform-Traversal	0:1	*xsd:boolean*	For Rail, whether this is a platform traversal at speed, typically triggering an announcement to stand back. Default is *'false'*. *DetailLevel*: normal.
	SignalStatus	0:1	*xsd:NMTOKEN*	Status of signal clearance for train. This may affect the presentation emphasis given to arrival or departures on displays – e.g. cleared trains appear first, flashing in green. *DetailLevel*: full.
Call Property	***TimingPoint***	0:1	*xsd:boolean*	Whether the stop is a TIMING POINT, i.e. times are measured at it. In Some systems this is a measure of data quality as non-timing points are interpolated. *DetailLevel*: full.
	BoardingStretch	0:1	*xsd:boolean*	Whether this is a Hail and Ride Stop. Default is *'false'*. *DetailLevel*: full.
	RequestStop	0:1	*xsd:boolean*	Whether VEHICLE stops only if requested explicitly by passenger. Default is *'false'*. *DetailLevel*: full.
	OriginDisplay	0:*	*NLString*	The name of the Origin of the journey; used to help identify the VEHICLE to the public: Since VEHICLEs can change their displayed origin during a journey, the Origin included here should be what the VEHICLE will display as it reaches this stop. (SIRI v2.0). One per language
	Destination-Display	0:*	*NLString*	The name of the destination of the journey; used to help identify the VEHICLE to the public. Since VEHICLEs can change their destination during a journey, the destination included here should be what the VEHICLE will display when it reaches this stop. One per language (Unbounded 0:* since +SIRI v2.0). *DetailLevel*: normal.
Call Note	***CallNote***	0:*	*NLString*	Text annotation that applies to this CALL. *DetailLevel*: full.

Disruption-Group	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***DisruptionGroup***. *DetailLevel*: normal.
Real-time Arrival times	***AimedArrival-Time***	0:1	*xsd:dateTime*	Arrival time of VEHICLE in either the original or Production Timetable. *DetailLevel*: minimum.
	Actual-ArrivalTime	0:1	*xsd:dateTime*	Observed time of arrival of VEHICLE. *DetailLevel*: minimum.
	ExpectedArrival-Time	0:1	*xsd:dateTime*	Estimated time of arrival of VEHICLE. *DetailLevel*: minimum.
	LatestExpectedArrivalTime	0:1	*xsd:dateTime*	Latest expected time at which a VEHICLE will arrive at stop. +SIRI v2.0
Arrival-Status	***:::***	0:1	*MonitoredStop-ArrivalStatus-Group*	See Below ***MonitoredStopArrivalStatusGroup***.
Departure Times	***AimedDeparture-Time***	0:1	*xsd:dateTime*	Departure time of VEHICLE in either the original or Production Timetable. *DetailLevel*: minimum.
	Actual-DepartureTime	0:1	*xsd:dateTime*	Actual observed time of departure of VEHICLE. *DetailLevel*: minimum.
	Expected-DepartureTime	0:1	*xsd:dateTime*	Estimated time of departure of VEHICLE to show to public. *DetailLevel*: minimum.
	Provisional-Expected-DepartureTime	0:1	*xsd:dateTime*	Estimated departure time of VEHICLE without waiting time due to operational actions. This would normally be shown to teh public at a stop if different from the Expected Departure time. +SIRI v2.0. *DetailLevel*: normal.
	EarliestExpected DepartureTime	0:1	*xsd:dateTime*	Earliest time at which VEHICLE may leave the stop. Used to secure connections. Passengers must be at boarding point by this time to be sure of catching VEHICLE. +SIRI v2.0. *DetailLevel*: normal.
	Expected-Departure-PredictionQuality	0:1	*+Prediction-Quality*	Prediction quality, either as approximate confidence level or as a more quantitative percentile range of predictions that will fall within a given range of times. See below ***ExpectedDeparturePredictionQuality*** +SIRI v2.0. If not defined for some calls, an Extrapolation Rule has to be applied, see 6.7. *DetailLevel*: full.
Passenger-Times	***AimedLatest-Passenger-AccessTime***	0:1	*xsd:dateTime*	Target latest time at which a PASSENGER should aim to arrive at the STOP PLACE containing the stop. This time may be earlier than the VEHICLE departure times and may include time for processes such as check-in, security, etc. (As specified by CHECK CONSTRAINT DELAYs in the underlying data) If absent assume to be the same as Earliest expected departure time. +SIRI v2.0
	ExpectedLatest-Passenger-AccessTime	0:1	*xsd:dateTime*	Expected latest time at which a PASSENGER should aim to arrive at the STOP PLACE containing the stop. This time may be earlier than the VEHICLE departure times and may include time for processes such as check-in, security, etc. (As specified by CHECK CONSTRAINT DELAYs in the underlying data) If absent assumed to be the same as Earliest expected departure time. +SIRI v2.0
Departure Status	***:::***	0:1	*MonitoredStop DepartureStatus-Group*	See Below ***MonitoredStopDepartureStatusGroup***.

Boarding	***AimedHeadway-Interval***	0:1	*Positive-DurationType*	For frequency based services, target interval between services at stop. *DetailLevel*: minimum.
	Expected-HeadwayInterval	0:1	*Positive-DurationType*	Estimated interval for frequency based service. *DetailLevel*: minimum.
Stop Proximity Group	***DistanceFrom-Stop***	0:1	*DistanceType*	Distance of VEHICLE from stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. Positive value denotes distance before stop. +SIRI v2.0.
	NumberOf-StopsAway	0:1	*nonNegative-Integer*	Count of stops along SERVICE PATTERN between current position of VEHICLE and stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. +SIRI v2.0.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.4.4.2 *ArrivalBoardingActivity* — Allowed Values

Allowed values for ***ArrivalBoardingActivity*** (*ArrivalBoardingActivity Enumeration*).

Table 48 — *ArrivalBoardingActivity* —Allowed Values

Value	Description
alighting	Passengers may alight .
noAlighting	Passengers may not alight.
passthru	Vehicle does not stop.

8.7.4.4.3 *DepartureBoardingActivity* — Allowed Values

Allowed values for ***DepartureBoardingActivity*** (*DepartureBoardingActivityEnumeration*).

Table 49 — *DepartureBoardingActivity* —Allowed Values

Value	Description
boarding	Passengers may board.
noAlighting	Passengers may not board.
passthru	Vehicle does not stop.

8.7.4.4.4 MonitoredStopArrivalStatus — Group

The ***MonitoredStopArrivalStatusGroup*** describes common arrival elements of a ***MonitoredCall*** and an ***OnwardCall.***

Table 50 — *MonitoredStopArrivalStatusGroup* — Elements

MonitoredStopArrivalStatusGroup			*+Group*	Elements relating to arrival status.
Arrival Status	***ArrivalStatus***	0:1	*CallStatusEnum*	Classification of the timeliness of the arrival part of the CALL according to a fixed list of values. This may reflect a presentation policy, for example CALLs less than one minute behind target time are still classified as on-time. Applications may use this to guide their own presentation of times. If not specified, same as DepartureStatus. *onTime* | *early* | *delayed* | *cancelled* | *arrived* | *departed* | *missed* | *noReport* See Part2 common definitions. *DetailLevel*: normal.
	ArrivalProximity-Text	0:*	*NLString*	Arbitrary text string to show to indicate the proximity status of the arrival of the VEHICLE , for example, *"Enroute", "5 Km",* "*Approaching*". May depend on the policy of the operator, for example show "*Approaching*" if less than 200metres away from stop. +SIRI v2.0. One per language *DetailLevel*: normal.
	Arrival-PlatformName	0:1	*NLString*	Bay or platform name. Inherited property. Can be omitted if the same as the ***DeparturePlatformName***. If there not an arrival platform name separate from the departure platform name, the precedence is (i) any arrival platform on any related dated timetable CALL element, (ii) any departure platform name on this CALL element; (iii) any departure platform name on any related dated timetable CALL. *DetailLevel*: basic.
	ArrivalBoarding-Activity	0:1	*ArrivalBoarding-Activity Enumeration*	Type of alighting activity allowed at stop. *alighting* | *noAlighting* | *passthru.* Default is '*alighting'*. *DetailLevel*: normal.
Arrival Stop Assignment	***ArrivalStop-Assignment***	0:1	*+Structure*	Assignment of arrival of Scheduled STOP POINT to a physical QUAY (platform). If not given, assume same as for departure +SIRI v2.0.
	Aimed-QuayRef	0:1	*→QuayCode-Type*	Physical QUAY to use according to the planned timetable. +SIRI v2.0
	Aimed-QuayName	0:1	*NLString*	Scheduled Platform name. Can be used to indicate a platform change. +SIRI v2.0.
	Expected-QuayRef	0:1	*→QuayCode-Type*	Physical QUAY to use according to the real-time prediction. +SIRI v2.0
	Actual-QuayRef	0:1	*→QuayCode-Type*	Physical QUAY actually used. +SIRI v2.0.

8.7.4.4.5 *ArrivalStatus/DepartureStatus* — Allowed Values

Allowed values for ***ArrivalStatus*** and ***DepartureStatus*** (*CallStatusEnumeration*).

Table 51 — *CallStatus* —Allowed Values

Value	Description
onTime	Vehicle is on time,
early	Vehicle is early,
delayed	Vehicle is delayed
cancelled	Arrival/Departure is cancelled
missed	Stop has been missed out but

	journey is not cancelled, +SIRI v2.0
arrived	Vehicle has arrived at stop
departed	Vehicle has departed stop. +SIRI v2.0
noReport	There is no report of status.
notExpected	Arrival/departure is not expected. A flexible service that is not (yet) preordered.

8.7.4.4.6 MonitoredStopDepartureStatus — Group

The ***MonitoredStopDepartureStatusGroup*** describes common departure elements of a ***MonitoredCall*** and an ***OnwardCall***.

Table 52 — *MonitoredStopDepartureStatusGroup* — Elements

MonitoredStopDepartureStatusGroup				*+Group*	Elements relating to departure status.
Departure Status		***DepartureStatus***	0:1	*CallStatusEnum*	Classification of the timeliness of the departure part of the CALL, according to a fixed list of values. This may reflect a presentation policy, for example CALLs less than one minute behind target time are still classified as on-time. Applications may use this to guide their own presentation of times. See Part2 common definitions. *onTime* | *early* | *delayed* | *cancelled* | *arrived* | *departed* | *missed* | *noReport* *DetailLevel*: normal.
		Departure-ProximityText	0:*	*NLString*	Arbitrary text string to show to indicate the proximity status of the departure of the VEHICLE, for example, *"Boarding", "GatesClosed".* +SIRI v2.0 One per language Detail Level: normal
		Departure-PlatformName	0:1	*NLString*	Bay or platform (QUAY) name from which VEHICLE will depart. Inherited property. *DetailLevel*: basic.
		Departure-BoardingActivity	0:1	*Departure-Boarding-ActivityEnum*	Type of boarding activity allowed at stop. Default is '*boarding'*. *boarding* | *noBoarding* | *passthru.* *DetailLevel*: normal.
Departure Stop Assignment		***DepartureStop-Assignment***	0:1	*+Structure*	Assignments of departure platform for SCHEDULED STOP POINT to a physical QUAY. +SIRI v2.0. +SIRI v2.0. *DetailLevel*: normal.
		Aimed-QuayRef	0:1	*→QuayCode-Type*	Physical QUAY (Platform) to use according to the planned timetable. +SIRI v2.0
		Aimed-QuayName	0:*	*NLString*	Scheduled QUAY (Platform) name. Can be used to indicate a platform change. +SIRI v2.0 One per language
		Expected-QuayRef	0:1	*→QuayCode-Type*	Physical QUAY (Platform) to use according to the real-time prediction. +SIRI v2.0
		Actual-QuayRef	0:1	*→QuayCode-Type*	Physical QUAY (Platform) actually used. +SIRI v2.0

8.7.4.5 MonitoredVehicleJourney / OnwardCall — Element

8.7.4.5.1 General

The ***OnwardCall*** element describes a CALL which has still to be made in the ***MonitoredVehicleJourney***.

It will only be populated for a ***MonitoredStopVisit*** if a ***StopMonitoringDetailLevel*** of *calls* is requested. The number of onward CALLs to include is given by the ***MaximumNumberOfCalls*** / ***Onward*** element.

Table 53 — MonitoredStopVisit / MonitoredVehicleJourney / OnwardCall — Attributes

OnwardCall			*+Structure*	Information on a CALL at a stop after the current CALL.
Stop Identity	***StopPointRef***	0:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT.
	VisitNumber	0:1	*VisitNumber-Type*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to the same stop, the ***VisitNumber*** is used to distinguish each separate visit.
	Order	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Name of Stop. One per language (Unbounded 0:* since +SIRI v2.0).
Progress	***VehicleAtStop***	0:1	*xsd:boolean*	Whether VEHICLE is at stop at the current time. Default is *'false'*.
	TimingPoint	0:1	*xsd:boolean*	Whether the stop is a TIMING POINT, i.e. times are measured at it. In Some systems this is a measure of data quality as non-timing points are interpolated.
Arrival Times	***AimedArrivalTime***	0:1	*xsd:dateTime*	Timetabled arrival time of VEHICLE in either the original or Production Timetable.
	ExpectedArrival-Time	0:1	*xsd:dateTime*	Estimated time of arrival of VEHICLE.
	ExpectedArrival-PredictionQuality	0:1	*+Prediction-Quality*	Prediction quality, either as approximate confidence level or as a more quantitative percentile range of predictions that will fall within a given range of times. Compare below ***ExpectedDeparturePredictionQuality*** +SIRI v2.0. If not defined for some CALLs, an Extrapolation Rule can be applied, see 6.7. *DetailLevel*: full.
Arrival Status	***:::***	0:1	*StopArrival-StatusGroup*	See above ***MonitoredStopArrivalStatusGroup***.
Departure Times	***:::***	0:1	*Onwards Departure TimesGroup*	See below ***OnwardsVehicleDepartureTimesGroup***
Departure Status	***:::***	0:1	*StopDeparture StatusGroup*	See above ***MonitoredStopDepartureStatusGroup***.
Progress Status	***AimedHeadWay-Interval***	0:1	*Positive-DurationType*	For frequency based services, target frequency of services at stop.
	ExpectedHeadway-Interval	0:1	*Positive-DurationType*	Estimated HEADWAY INTERVAL between VEHICLEs for frequency based service.

Stop Proximity	***DistanceFromStop***	0:1	*DistanceType*	Distance of VEHICLE from stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. Positive value denotes distance before stop. +SIRI v2.0.
	NumberOf-StopsAway	0:1	*nonNegative-Integer*	Count of stops along SERVICE PATTERN between current position of VEHICLE and stop of CALL as measured along ROUTE track. Only shown if detail level is 'calls' or higher. +SIRI v2.0.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

The **OnwardCall** doesn't have any ActualArrivalTime or ActualDepartureTime since the vehicle usually doesn't have reached the correspondin stops yet. However, it is possible to create a request with a **StartTime** in the past. In suche situation ActualArrivalTime and ActualDepartureTime will be provided in EstimatedArrivalTime and EstimatedDepartureTime (being the las estimated values) and the status of the call will be set to "arrived" or "departed".

8.7.4.5.2 OnwardsVehicleDepartureTimesGroup

The ***OnwardsVehicleDepartureTimesGroup*** describes a set of onwards departure times at a given stop at which the vehicle has not yet arrived.

Table 54 — OnwardCall / *OnwardsVehicleDepartureTimesGroup* — Attributes

Onwards Vehicle Departure Time	***AimedDeparture-Time***	0:1	*xsd:dateTime*	Timetabled departure time of the VEHICLE in either the original or Production Timetable.
	ExpectedDeparture-Time	0:1	*xsd:dateTime*	Estimated time of departure of the VEHICLE from the onwards timing point..
	ProvisionalExpected DepartureTime	0:1	*xsd:dateTime*	Estimated departure time of VEHICLE without waiting time due to operational actions. For people at stop this would normally be shown if different from Expected departure time. +SIRI v2.0.
	EarliestExpected-DepartureTime	0:1	*xsd:dateTime*	Earliest time at which VEHICLE may leave the stop. Used to secure connections. Used for passenger announcements. Passengers must be at boarding point by this time to be sure of catching VEHICLE. +SIRI v2.0
	ExpectedDeparture-PredictionQuality	0:1	*+Prediction-Quality*	Prediction quality, either as approximate confidence level or as a more quantitative percentile range of predictions that will fall within a given range of times. See below ***ExpectedDeparturePredictionQuality*** +SIRI v2.0. If not defined for some CALLs, an Extrapolation Rule can be applied, see 6.7. *DetailLevel*: full.

8.7.4.5.3 *ExpectedDeparturePredictionQuality* — Element

The ***ExpectedDeparturePredictionQuality*** indicates the quality of a prediction. A default quantitative measures may be given (see discussion of prediction quality in 6.7), or an override value specific to this CALL. Successive values may be interporlated.

Table 55 — OnwardCall / *ExpectedDeparturePredictionQuality* — Attributes

StopLineNotice			*+Structure*	Line NOTICE for a stop.
Value	***PredictionLevel***	**1:1**		An approxiimate characterisation of prediction quality as one of five values. +SIRI v2.0. *certain* | *veryReliable* | *relative* | *probablyReliable* | *unconfirmed* See discussion of prediction quality in 6.7 for default bounds associated sith confidence levels.
Range	***Percentile***	0:1	*decmal*	Default percentile for which predictions fall within lower and upper limits, defaulut is 90%
	LowerTime-Limit	**0:1**	*xsd:dateTime*	Lower bound on time of prediction specific +SIRI v2.0
	UpperTimeLimit	**0:1**	*xsd:dateTime*	Upper bound on time of prediction. +SIRI v2.0

8.7.5 MonitoredStopVisitCancellation — Element

The ***MonitoredStopVisitCancellation*** element is used to cleardown an earlier ***MonitoredStopVisit***, i.e. to indicate that it can be removed from a display. This is useful in particular if only updates are being exchanged.

NOTE Direct cleardown can also be used to provide a fast way of identifying VEHICLEs that have arrived, using vehicle-to-stop radio communication.

A ***MonitoredStopVisitCancellation*** shall contain sufficient information to identify the corresponding earlier ***MonitoredStopVisit*** entry; either in the form of an explicit reference (*ItemRef*), or by reference values sufficient to uniquely distinguish the cleared Stop Visit from other visits.

Table 56 — MonitoredStopVisitCancellation — Attributes

MonitoredStopVisitCancellation			*+Structure*	Cancellation of a previously communicated ***StopVisit.***
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which ***MonitoredStopVisitCancellation*** was recorded.
Event-Identity	***ItemRef***	0:1	*ItemIdentifier*	Reference to a previous ***MonitoredStopVisit*** which this item cancels.
	MonitoringRef	**1:1**	*→Monitoring¬Code*	Reference to a Stop Monitoring point (LOGICAL DISPLAY) at which visits happen. May be a SCHEDULED STOP POINT or a display identifier.
	VisitNumber	0:1	*VisitNumberType*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to a stop, the ***VisitNumber*** is used to distinguish each separate visit.
	ClearDownRef	0:1	*→ClearDownCode*	Identifier associated with ***StopVisit*** for use in direct wireless communication between VEHICLE and stop display. Cleardown codes are short arbitrary *identifiers* suitable for radio transmission.
	LineRef	0:1	*→LineCode*	Reference to a LINE of journey that is being deleted.
	DirectionRef	0:1	*→DirectionCode*	Reference to a DIRECTION of journey that is being deleted.
	Vehicle-JourneyRef	0:1	*+Structure*	Reference to a *DATED* VEHICLE JOURNEY that is being deleted. *See SIRI Part 2.*
	DataFrame-Ref	**1:1**	*+ DataFrameRef*	Reference to a Data Frame, within which the given *DATED* VEHICLE reference is unique

	Vehicle-JourneyRef	**1:1**	*FramedVehicle-JourneyRef*	Reference to a *DATED* VEHICLE JOURNEY that is being deleted.
Journey-Pattern-Info	***:::***	0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Message	***Reason***	0:*	*NLString*	Reason for cancellation. One per language (Unbounded 0:* since +SIRI v2.0).
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

8.7.6 StopLineNotice — Element

Each ***StopLineNotice*** describes a text notice relevant to a SCHEDULED STOP POINT about a LINE.

Each ***StopLineNotice*** has an identifier: this can be used to reference events when sending incremental updates – for example the ***StopLineNotice*****Cancellation** indicates a previous element has been removed.

Table 57 — StopLineNotice — Attributes

StopLineNotice			*+Structure*	Line NOTICE for a stop.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded. *DetailLevel*: minimum.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used to refer to item subsequently for server side cleardown. *DetailLevel*: minimum.
	MonitoringRef	**1:1**	*→Monitoring¬Code*	Reference to a Stop Monitoring point including LINE to which NOTICE applies. May be a SCHEDULED STOP POINT or a display identifier. *DetailLevel*: minimum.
Route Identity	***LineRef***	**1:1**	*→LineCode*	Reference to a LINE. *DetailLevel*: minimum.
	DirectionRef	**1:1**	*→DirectionCode*	Identifier of DIRECTION the VEHICLE is running along the JOURNEY PATTERN, for example, "in" or "out", “clockwise”. Distinct from a DESTINATION. *DetailLevel*: minimum.
	PublishedLine-Name	0:*	*NLString*	Name or Number by which the LINEis known to the public. +SIRI v2.0. *DetailLevel*: basic.
Message	***LineNote***	**1:1**	*+Structure*	NOTICE associated with delivery. *DetailLevel*: basic.
	DeliveryVariant	0:*	*+Structure*	Alternative versions of the text of the NOTICE for use in particular circumstances. +SIRI v2.0. *DetailLevel*: normal.
	VariantType	0:1	*xsd:normalized-String*	Classification of variant. +SIRI v2.0.
	Content	**1:1**	*NLString*	Variant text. +SIRI v2.0.
Situation	***SituationRef***	0:*	*→SituationCode*	Reference to a SITUATION associated with the element. *DetailLevel*: normal.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.7 StopLineNoticeCancellation — Element

The ***StopLineNoticeCancellation*** element is used to cleardown an earlier ***StopLineNotice***, i.e. to indicate that a notice is no longer current and can be removed from a display.

A ***StopLineNoticeCancellation*** shall contain sufficient information to identify the corresponding earlier ***StopLineNotice*** entry; either in the form of an explicit reference (***ItemRef***), or by reference values sufficient to uniquely distinguish the cleared ***StopLineNotice*** from other instances.

Table 58 — StopLineNoticeCancellation — Attributes

StopLineNoticeCancellation			*+Structure*	Cancellation of an earlier *StopLineNotice*.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which cancellation was recorded.
Event Identity	***ItemRef***	0:1	*→ItemIdentifier*	Reference to a previous ***StopLineNotice*** which this cancels.
	MonitoringRef	**1:1**	*→MonitoringCode*	Reference to a Stop Monitoring point.
	LineRef	**1:1**	*→LineCode*	Reference to a LINE.
	DirectionRef	**1:1**	*→DirectionCode*	Identifier of DIRECTION the VEHICLE is running along the JOURNEY PATTERN, for example, "in" or "out", “clockwise”. Distinct from a DESTINATION
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.8 StopNotice — Element (+SIRI v2.0)

Each ***StopNotice*** describes a notice relevant to a Stop Monitoring point by referencing a SITUATION.

Each ***StopNotice*** has an identifier: this can be used to reference events when sending incremental updates – for example the ***StopNotice*****Cancellation** indicates a previous element has been removed.

Table 59 — StopNotice — Attributes

StopNotice			*+Structure*	NOTICE for a stop.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded. *DetailLevel*: minimum.
Identity	***ItemIdentifier***	**1:1**	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used to refer to item subsequently for server side cleardown. *DetailLevel*: minimum.
	MonitoringRef	**1:1**	*→Monitoring-Code*	Reference to a Stop Monitoring point where NOTICE applies. May be a SCHEDULED STOP POINT or a display identifier. *DetailLevel*: minimum.
Stop Point Identity	***StopPointRef***	**1:1**	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT.
Situation Identity	***SituationRef***	**1:1**	*+Structure*	Reference to a SITUATION associated with the element.
	Situation-SimpleRef	0:1	*→Situation-Code*	Reference to a SITUATION associated with the element.
	Situation-FullRef	0:1	*→Situation-Code*	Reference to a SITUATION associated with the element.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.9 StopNoticeCancellation — Element (+SIRI v2.0)

The ***StopNoticeCancellation*** element is used to cleardown an earlier ***StopNotice***, i.e. to indicate that a notice is no longer current and can be removed from a display.

A ***StopNoticeCancellation*** shall contain sufficient information to identify the corresponding earlier ***StopNotice*** using an explicit reference (***ItemRef***).

Table 60 — StopNoticeCancellation — Attributes

StopNoticeCancellation			*+Structure*	Cancellation of an earlier *StopNotice*.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which cancellation was recorded.
Event Identity	***ItemRef***	1:1	*→ItemIdentifier*	Reference to a previous ***StopNotice*** which is cancelled.
	MonitoringRef	**1:1**	*→MonitoringCode*	Reference to a Stop Monitoring point.
	StopPointRef	0:1	*→StopPointCode*	Reference to a SCHEDULED STOP POINT.
	Applies-FromTime	0:1	*xsd:dateTime*	Used to report a future cancellation.

8.7.10 ServiceException — Element (+SIRI v2.0)

8.7.10.1 General

The ***ServiceException*** element is used to give information about exceptions at the Stop or for some services at the stop.

Table 61 — ServiceException — Attributes

ServiceException			*+Structure*	Information about why data is unavailable for the functional service.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which exception was recorded.
Scope	***LineRef***	0:1	*→LineCode*	Reference to a LINE to which Exception applies.
	DirectionRef	0:1	*→DirectionCode*	Reference to a DIRECTION of LINE to which Exception applies.
	StopPointRef	0:1	*→StopPointCode*	Reference to a SCHEDULED STOP POINT to which Exception applies.
	Service-Exception-Status	0:1	*ServiceException-Enum*	Status of Service exception
	Notice	0:*	*NLString*	Notice associated with Service Exception. One per language (Unbounded 0:* since +SIRI v2.0).
	SituationRef	0:1	*→SituationRef*	Reference to a SITUATION providing further details about the Exception.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

8.7.10.2 *ServiceExceptionStatus* — Allowed Values (+SIRI v2.0)

ServiceExceptionStatus provides the status of service: Service not yet started, Service ended for day, no service today, etc.

Table 62 — *ServiceExceptionStatus* — Allowed Values

Name	Definition
beforeFirstJourney	No transport services returned because currently before first journey of day.
afterLastJourney	No transport services returned because currently after last journey of day.
noServiceToday	No transport services returned because no services today.
transportTemporarilySuspended	No transport services returned because services currently suspended.
transportLongtermSuspended	No transport services returned because of prolonged suspension of services.
transportSeverlyDisrupted	Transport services returned subject to severe disruptions.
realtimeDataNotAvailable	No transport services returned because real-time services not available.

8.7.11 StopMonitoringDelivery — Examples

8.7.11.1 StopMonitoringDelivery — Basic Example

The following is an example of a ***StopMonitoringDelivery***. It shows a single Stop Visit for a detail level of *Normal*

```
<ServiceDelivery>
        <!--=======HEADER================================================ -->
        <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
        <ProducerRef>KUBRICK</ProducerRef>
        <Status>true</Status>
        <MoreData>false</MoreData>
        <!--=======PAYLOAD================================================ -->
        <StopMonitoringDelivery version="0.1d">
            <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
            <SubscriberRef>NADER</SubscriberRef>
            <SubscriptionRef>2004-12-17T09:30:47-05:00</SubscriptionRef>
            <Status>true</Status>
            <ValidUntil>2004-12-17T09:30:47-05:00</ValidUntil>
            <ShortestPossibleCycle>PT3M</ShortestPossibleCycle>
            <!--====FIRST ARRIVAL ============================================ -->
            <MonitoredStopVisit>
                <RecordedAtTime>2004-12-17T09:25:46-05:00</RecordedAtTime>
                <MonitoringRef>HLTST011</MonitoringRef>
                <MonitoredVehicleJourney>
                    <!-- JOURNEY IDENTITY GROUP -->
                    <LineRef>Line123</LineRef>
                    <DirectionRef>Out</DirectionRef>
                    <FramedVehicleJourneyRef>
                        <DataFrameRef>2004-12-17</DataFrameRef>
                        <DatedVehicleJourneyRef>034567</DatedVehicleJourneyRef>
                    </FramedVehicleJourneyRef>
                    <!-- JOURNEY PATTERN INFO GROUP -->
                    <PublishedLineName>123</PublishedLineName>
                    <DestinationName>Paradise Park</DestinationName>
                    <VehicleRef>VEH987654</VehicleRef>
                    <MonitoredCall>
                        <VisitNumber>0014</VisitNumber>
                        <VehicleAtStop>false</VehicleAtStop>
                        <!-- STOP MONITORING TIMES GROUP -->
                        <AimedArrivalTime>2004-12-17T09:40:46-05:00</AimedArrivalTime>
                        <ExpectedArrivalTime>2004-12-17T09:40:46-05:00</ExpectedArrivalTime>
                        <AimedDepartureTime>2004-12-17T09:42:47-05:00</AimedDepartureTime>
                        <ExpectedDepartureTime>2004-12-17T09:40:47-05:00</ExpectedDepartureTime>
                    </MonitoredCall>
                    <NextStopPointRef>HLTST012</NextStopPointRef>
                </MonitoredVehicleJourney>
```

```
        </MonitoredStopVisit>
    </StopMonitoringDelivery>
</ServiceDelivery>
```

8.7.11.2 StopMonitoringDelivery — Extended Example

The following is an example of a ***StopMonitoringDelivery*** with several cancellations as well as a more verbose level of detail.

```
<ServiceDelivery>
    <!--=======HEADER==================================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <MoreData>false</MoreData>
    <!--=======PAYLOAD================================================ -->
    <StopMonitoringDelivery version="0.1d">
        <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
        <SubscriberRef>NADER</SubscriberRef>
        <SubscriptionRef>2004-12-17T09:30:47-05:00</SubscriptionRef>
        <Status>true</Status>
        <ValidUntil>2004-12-17T09:30:47-05:00</ValidUntil>
        <ShortestPossibleCycle>PT3M</ShortestPossibleCycle>

        <!--=====FIRST ARRIVAL ============================================ -->
        <MonitoredStopVisit>
            <RecordedAtTime>2004-12-17T09:25:46-05:00</RecordedAtTime>
            <!-- IDENTITY GROUP -->
            <ItemIdentifier >SED9843214675432</ItemIdentifier >
            <MonitoringRef>HLTST011</MonitoringRef>
            <ClearDownRef>CLR7654</ClearDownRef>
            <MonitoredVehicleJourney>
                <!-- JOURNEY IDENTITY GROUP -->
                <LineRef>Line123</LineRef>
                <DirectionRef>NMTOKEN</DirectionRef>
                <FramedVehicleJourneyRef>
                    <DataFrameRef>2004-12-17</DataFrameRef>
                    <DatedVehicleJourneyRef>98764</DatedVehicleJourneyRef>
                </FramedVehicleJourneyRef>
                <!-- JOURNEY PATTERN INFO GROUP -->
                <PublishedLineName>123</PublishedLineName>
                <!-- SERVICE INFO GROUP -->
                <OperatorRef>OP22</OperatorRef>
                <ProductCategoryRef>PDCATEXPRESS</ProductCategoryRef>
                <ServiceFeatureRef>SERVCCAT551</ServiceFeatureRef>
                <!-- SERVICE POINTS GROUP -->
                <OriginRef>PLACE21</OriginRef>
                <OriginName>Highbury</OriginName>
                <Via>
                    <PlaceName>Kensall Green</ PlaceName>
                </Via>
                <Via>
                    <PlaceName>Roman Road</PlaceName>
                </Via>
                <DestinationRef>PLACE45</DestinationRef>
                <DestinationName>Paradise Park</DestinationName>
                <!-- JOURNEY INFO GROUP -->
                <JourneyNote>Kensall Green</JourneyNote>
                <!--JOURNEY PROGRESS GROUP -->
                <Monitored>true</Monitored>
                <InCongestion>false</InCongestion>
                <VehicleLocation>
                    <Longitude>180</Longitude>
                    <Latitude>90</Latitude>
                </VehicleLocation>
                <Delay>PT2M</Delay>
```

```
            <ProgressStatus> onTime</ProgressStatus>
            <!-- Operational Info GROUP -->
            <TrainBlockPart>
                <NumberOfBlockParts>1</NumberOfBlockParts>
                <TrainPartRef>3456</TrainPartRef>
                <PositionOfTrainBlockPart>1</PositionOfTrainBlockPart>
            </TrainBlockPart>
            <BlockRef>BLOCK765</BlockRef>
            <CourseOfJourneyRef>RUN765</CourseOfJourneyRef>
            <VehicleRef>VEH987654</VehicleRef>
            <!-- MONITORED CALLING PATTERN GROUP -->
            <PreviousCalls>
                <PreviousCall>
                    <StopPointRef>HLT0010</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>
            <MonitoredCall>
                <VisitNumber>0014</VisitNumber>
                <!-- STOP PROGRESS GROUP -->
                <VehicleAtStop>false</VehicleAtStop>
                <VehicleLocationAtStop>
                    <Longitude>180</Longitude>
                    <Latitude>90</Latitude>
                </VehicleLocationAtStop>
                <!-- STOP MONITORING TIMES GROUP -->
                <AimedArrivalTime>2004-12-17T09:40:46-05:00</AimedArrivalTime>
                <ExpectedArrivalTime>2004-12-17T09:40:46-05:00</ExpectedArrivalTime>
                <AimedDepartureTime>2004-12-17T09:42:47-05:00</AimedDepartureTime>
                <ExpectedDepartureTime>2004-12-17T09:40:47-05:00</ExpectedDepartureTime>
                <DeparturePlatformName>Bay 5</DeparturePlatformName>
                <!-- MONITORED VEHICLE JOURNEY -->
            </MonitoredCall>
            <OnwardCalls>
                <OnwardCall>
                    <StopPointRef>HLTST012</StopPointRef>
                    <VisitNumber>4</VisitNumber>
                    <StopPointName>String</StopPointName>
                    <VehicleAtStop>false</VehicleAtStop>
                    <AimedArrivalTime>2004-12-17T09:30:56-05:00</AimedArrivalTime>
                    <ExpectedArrivalTime>2004-12-17T09:30:56-05:00</ExpectedArrivalTime>
                    <AimedDepartureTime>2004-12-17T09:30:57-05:00</AimedDepartureTime>
                    <ExpectedDepartureTime>2004-12-17T09:30:57-05:00</ExpectedDepartureTime>
                </OnwardCall>
            </OnwardCalls>
        </MonitoredVehicleJourney>
        <StopVisitNote>Hello bus for line 123 !</StopVisitNote>
    </MonitoredStopVisit>
    <!--====Second ARRIVAL ====This Vehicle is at the stop============================== -->
    <!--====FIRST DELETION OF EARLIER ARRIVAL =====BY SYSTEM KEY================ -->
    <MonitoredStopVisitCancellation>
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
        <ItemRef>SED9843214675429</ItemRef>
        <Reason>Arrived</Reason>
    </MonitoredStopVisitCancellation>
    <!--====Second DELETION OF EARLIER ARRIVAL ==IDENTIFIERS ARE REF DATA======== -->
    <MonitoredStopVisitCancellation>
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
        <MonitoringRef>HLTST113</MonitoringRef>
        <VisitNumber>2</VisitNumber>
        <LineRef>Line123</LineRef>
        <DirectionRef>Out</DirectionRef>
        <FramedVehicleJourneyRef>
            <DataFrameRef>2004-12-17</DataFrameRef>
```

```
                <DatedVehicleJourneyRef>0987656</DatedVehicleJourneyRef>
            </FramedVehicleJourneyRef>
            <Reason>Arrived</Reason>
        </MonitoredStopVisitCancellation>
        <StopLineNotice>
            <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
            <ItemIdentifier >SED9843214675429</ItemIdentifier >
            <MonitoringRef>HLTST113</MonitoringRef>
            <LineRef>123</LineRef>
            <DirectionRef>123</DirectionRef>
            <LineNote>What, will the line stretch out to the crack of doom?</LineNote>
        </StopLineNotice>
        <StopLineNoticeCancellation>
            <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
            <ItemRef>SED9843214675429</ItemRef>
            <MonitoringRef>HLTST1143</MonitoringRef>
            <LineRef>123</LineRef>
            <DirectionRef>123</DirectionRef>
        </StopLineNoticeCancellation>
        <Note>Hello Stop</Note>
    </StopMonitoringDelivery>
</ServiceDelivery>
```

8.7.11.3 StopLineNotice — Example

The following is an example of a ***StopLineNotice*** and a ***StopLineNoticeCancellation****:*

```
<StopMonitoringDelivery version="1.0">>
    <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
    <MonitoringRef>HLTST011</MonitoringRef>
    <!--=====FIRST DELETION OF EARLIER ARRIVAL ==BY SYSTEM ASSIGNED ID== -->
    <StopLineNotice>
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
        <ItemRef>SED9843214675429</temRef>
        <LineRef>123</LineRef>
        <DirectionRef>123</ DirectionRef>
        <LineNote>What, will the line stretch out to the crack of doom?</LineNote>
    </ StopLineNotice>
    <MonitoringRef>HLTST011</MonitoringRef>
    <!--===== DELETION OF EARLIER ARRIVAL ==BY SYSTEM ASSIGNED ID== -->
    <StopLineNoticeCancellation>
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
        <ItemRef>SED9843214675429</ItemRef>
    </StopLineNoticeCancellation>
    <!--===== DELETION OF EARLIER ARRIVAL ==BY Reference== -->
    <StopLineNoticeCancellation >
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
        <LineRef>123</LineRef>
        <DirectionRef>123</ DirectionRef>
    </StopLineNoticeCancellation >
</StopMonitoringDelivery>
```

8.8 Using the Stop Timetable & Stop Monitoring services together

The Stop Monitoring service can be used in conjunction with the Stop Timetable service. The Stop Timetable service can be used to provision consumer systems with stop centric planned timetables. The Real Time monitoring system can then be used to supply deviations from the planned timetable.

9 Vehicle Monitoring Service [VM]

9.1 Purpose

The Vehicle Monitoring Service reports the position of a VEHICLE or group of VEHICLEs making monitored journeys in real-time. It can be used to monitor the progress of VEHICLEs, to provide information for systems which present visualisations of the movement of VEHICLEs, for example on maps, lists or line diagrams, and to exchange information about roaming VEHICLEs with other CONTROL CENTREs.

The Vehicle Monitoring Service comprises the ***VehicleMonitoringRequest*** message used to specify the contents of request or subscription messages, and the ***VehicleMonitoringDelivery*** message used to deliver the response. The ***VehicleMonitoringSubscription*** message allows a subscriber to request asynchronous updates for the service: it contains an embedded ***VehicleMonitoringRequest,*** along with further parameters controlling the asynchronous delivery.

VehicleMonitoringRequest has topic parameters to filter the information by VEHICLE, LINE, and DIRECTION, etc, and policy parameters to control the amount of data returned.

The ***VehicleMonitoringDelivery*** returns information about one or more VEHICLEs, each has: a ***MonitoredVehicleJourney*** instance, which may be populated to different levels of detail depending on the application requirements. The level of detail returned may range from very limited position information suitable for reporting just the relative position of a VEHICLE and its predicted arrival at the next stop, to a fully populated calling pattern, giving its predicted arrival over the whole journey, with section identifiers. The service can also be used to delivery historic record of all the observed times of a journey.

The ***MonitoredVehicleJourney*** allows the CALLs of a journey to be partitioned into three groups relative to the VEHICLE's current position, each of which may have slightly different properties:

a) CALLs already made, prior to the VEHICLE's current position;

b) the current CALL (which is defined as the stop at which the VEHICLE is parked or has most recently left;

c) onwards CALLs.

9.2 Reference Data

VehicleMonitoringRequest requires the participants to have agreed data reference models for:

a) Vehicle Monitoring Scopes;

b) LINEs;

c) DIRECTION.

Vehicle Monitoring Scopes are identified by a ***VehicleMonitoringRef*** and prescribe a pre-arranged arbitrary grouping of PTV services that may be referenced on requests.

9.3 Capability and Permission Matrices

9.3.1 Capability Matrix

The following set of required and optional capabilities is defined for the Vehicle Monitoring service. If the service supports Capability Discovery the ***VehicleMonitoringCapabilitiesRequest*** / ***VehicleMonitoring-CapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 63 — VehicleMonitoringCapabilities Matrix

VehicleMonitoringCapabilities			*+Structure*	Capabilities describing implementation of Vehicle Monitoring service.
inherit	***:::***	*0:1*	*See xxx-Capability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***	0:1	*+Structure*	Which optional filtering features are supported?
	DefaultPreviewInterval	**1:1**	*Positive-DurationType*	Default Preview Interval.
	FilterByVehicle-MonitoringRef	**1:1**	*xsd:boolean*	Whether results can be filtered by Monitoring point. Required Capability: Fixed as true.
	FilterByVehicleRef	0:1	*xsd:boolean*	Whether results can be filtered by ***VehicleRef***. Required Capability: Default is *'true'*.
	FilterByLineRef	0:1	*xsd:boolean*	Whether results can be filtered by ***LineRef***. Required Capability: Default is *'true'*.
	FilterByDirectionRef	0:1	*xsd:boolean*	Whether results can be filtered by ***DirectionRef***, Default is *'true'*.
Request Policy	***RequestPolicy***	0:1	*+Structure*	Which features of ***RequestPolicy*** are supported by service?
	Language	**1:***	*xsd:language*	National languages used by service.
	Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
	GmlCoordinateFormat	**1:1**	*SrsNameType*	Default coordinate format is given by a GML value.
	WgsDecimalDegrees		*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
	HasDetailLevel	0:1	*xsd:boolean*	Whether *DetailLevel* filtering is supported. Default *false*.
	DefaultDetailLevel	0:1	*Vehicle-Monitoring-DetailLevel-Enum*	Default *DetailLevel* if none specified on request. Default is 'n*ormal'*. *minimum* | *basic* | *normal* | *calls* | *full*
	HasMaximumVehicles	0:1	*xsd:boolean*	Whether results can be limited to a maximum number. Default is *'true'*.
	HasNumberOf-OnwardsCalls	0:1	*xsd:boolean*	If system can return detailed calling pattern, whether a number of onwards CALLs to include can be specified. Default is *'false'*.
	HasNumberOf-PreviousCalls	0:1	*xsd:boolean*	If system can return detailed calling pattern, whether a number of previous CALLs to include can be specified. Default is *'false'*.
Subscription-Policy	***SubscriptionPolicy***	0:1	*+Structure*	Which features of ***SubscriptionPolicy*** are supported by service?
	HasIncremental-Updates	0:1	*xsd:boolean*	Whether incremental updates can be specified for updates. Default is *'true'*.
	HasChangeSensitivity	0:1	*xsd:boolean*	Whether change threshold can be specified for updates. Default is *'true'*.

Access Control	***AccessControl***		0:1	*+Structure*	Which optional Access Control features are supported by service?
		RequestChecking	**1:1**	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
		CheckOperatorRef	0:1	*xsd:boolean*	If access control is supported, whether access control by OPERATOR is supported. Default is *'true'*.
		CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
		CheckVehicle-MonitoringRef	0:1	*xsd:boolean*	If access control is supported, whether access control by monitoring point is supported. Default is *'true'*.
Response	***ResponseFeatures***		0:1	*+Structure*	Which features of Response data are supported by service?
		HasLocation	0:1	*xsd:boolean*	Whether service supports Vehicle Location. Default is *'true'*.
any	***Extensions***		0:1	*any*	Placeholder for user extensions.

9.3.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***VehicleMonitoring-CapabilitiesResponse*** response can include the access permissions for the requestor participant to access data.

Table 64 — VehicleMonitoring Service — Permissions

VehicleMonitoringPermission			*+Structure*	Permissions to use implementation of Vehicle Monitoring service.
Inherit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant. See Part 2.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant. See Part 2.
	VehicleMonitoringPermissions	0:1	*+Structure*	Vehicle Monitoring reference permissions for participant. See Part 2

9.4 VehicleMonitoringRequest

9.4.1 VehicleMonitoringRequest — Element

9.4.1.1 General

The ***VehicleMonitoringRequest*** can be used in both a direct request, and for a subscription. If used for a subscription, additional Subscription Policy parameters shall apply.

Vehicles can be requested by LINE identifier to return all the VEHICLEs on a LINE, by area, or a combination of both. It is also possible to request the movement for a given VEHICLE.

The primary Topic term is the Vehicle for which information is to be returned.

Table 65 — VehicleMonitoringRequest — Attributes

VehicleMonitoringRequest			*+Structure*	Request for information about VEHICLE movements.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Vehicle Monitoring Service.
Endpoint Properties	***RequestTimestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	MessageIdentifier	0:1	*MessageQualifier*	
Topic	***VehicleMonitoringRef***	0:1	*→VehicleMonitoringFIlterCode*	The pre-arranged identifier about which data is requested.
			choice	One of the following:
	a ***VehicleRef***	0:1	*→VehicleCode*	Reference to a specific VEHICLE about which data is requested. Optional SIRI capability: *ByVehicle.*
	b ***LineRef***		*→LineCode*	Filter the results to include only VEHICLEs for the given LINE.
	DirectionRef	0:1	*→DirectionCode*	Filter the results to include only VEHICLEs going to the specified DIRECTION. Optional SIRI capability: *FilterByDirectionRef.*
Request Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *Language.*
	IncludeTranslations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
	MaximumVehicles	0:1	*xsd:positiveInteger*	The maximum number of VEHICLE JOURNEYs to include in a given delivery. The most recent n ***VehicleActivity*** instances within the look-ahead window are included. If absent, no limit.
	MaximumNumberOfCalls	0:1	*xsd:positiveInteger*	The maximum number of CALLs to include per MONITORED VEHICLE JOURNEY in a given delivery. Only applies if Detail is *calls*. +SIRI v2.0.
	VehicleMonitoringDetailLevel	0:1	*VehicleMonitoringDetailLevelEnum*	Level of detail to include in response. *minimum* | *basic* | *normal* | *calls* | *full* Default is '*normal*'. Optional SIRI capability: *DetailLevel* (if absent, must support *normal*).
	IncludeSituations	0:1	*xsd:boolean*	Whether any related Situations should be included in the ServiceDelivery. Default is 'false'. +SIRI v2.0
	MaximumNumberOfCalls	0:1	*+Structure*	If CALLs are to be returned in the response, maximum number of CALLs to include in response. If absent, include all CALLs. Optional SIRI capability: *DetailLevel*: calls.
	Previous	0:1	*xsd:positiveInteger*	Maximum number of previous CALLs to include. If set to 1, only the previous CALLs, if any is returned.
	Onwards	0:1	*xsd:positiveInteger*	Maximum number of onwards CALLs to include.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

9.4.1.2 VehicleMonitoringDetailLevel — Allowed Values

Different consumers will have different requirements for data. The *VehicleMonitoringDetailLevel* parameter on the request can be used to control how much detail is returned for each VEHICLE movement: see Table 61 below.

Table 66 — VehicleMonitoringDetailLevel — Allowed Values

Detail Level	Definition
minimum	Return only minimum data.
basic	Return useful basic minimum data.
normal	Return additional information about stop and destination.
callings	Return additional information including full calling pattern. Useful for trains.
full	Return all information including full calling pattern.

9.4.2 VehicleMonitoringRequest — Example

The following is an example of a ***VehicleMonitoringRequest***.

```
<ServiceRequest">
    <!--======ENDPOINT REFERENCES================================-->
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <VehicleMonitoringRequest version="1.0”>
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--=======TOPIC ==================================== -->
        <VehicleMonitoringRef>VEHPT55</VehicleMonitoringRef>
        <DirectionRef>Out</DirectionRef>
        <!--=======POLICY=============================================-->
        <VehicleMonitoringDetailLevel>Normal</VehicleMonitoringDetailLevel>
    </VehicleMonitoringRequest>
</ServiceRequest>
```

9.5 VehicleMonitoringSubscriptionRequest

9.5.1 VehicleMonitoringSubscriptionRequest — Element

The ***VehicleMonitoringSubscriptionRequest*** requests the asynchronous delivery of the information described by a ***VehicleMonitoringRequest***. The ***VehicleMonitoringSubscriptionRequestPolicy*** parameters control the processing of the subscription.

Table 67 — VehicleMonitoringSubscriptionRequest — Attributes

VehicleMonitoring-SubscriptionRequest			*+Structure*	Request for a subscription to the Vehicle Monitoring Service.
Identity	***SubscriberRef***	0:1	*→Participant-Code*	See SIRI Part 2 Common ***SubscriptionRequest*** parameters.
	Subscription-Identifier	**1:1**	*Subscription-Qualifier*	
Lease	***InitialTermination-Time***	**1:1**	*xsd:dateTime*	
Request	***VehicleMonitoring-Request***	**1:1**	*+Structure*	See ***VehicleMonitoringRequest***.

Policy		***Incremental-Updates***	0:1	*xsd:boolean*	Whether the producer should only provide updates to the last data returned, i.e. additions, modifications and deletions, or always return the complete set of current data. Default is *'true'*, i.e. once the initial transmission has been made, return only incremental updates. If *false* each subscription response will contain the full information as specified in this request. Optional SIRI capability: *IncrementalUpdates*.
				choice	One of the following:
	a	***ChangeBefore-Updates***	0:1	*Positive-DurationType*	The amount of change to the arrival or departure time that can happen before an update is sent (i.e. if ***ChangeBeforeUpdate*** is set to 2 min, the subscriber will not be told that a VEHICLE is 30 s delayed – an update will only be sent when the VEHICLE is at least 2 min delayed) Optional SIRI capability: *ChangeSensitivityThreshold*.
	b	***UpdateInterval***	0:1	*Positive-DurationType*	Time interval in seconds in which new data is to be transmitted. If unspecified, default to system configuration. Mutually exclusive with ***ChangeBeforeUpdates***.
		Extensions	0:1	xsd:any*	Placeholder for user extensions.

9.5.2 Sensitivity Threshold

When a subscription is first made, data for the current Vehicles being monitored is transmitted, thereafter, in order to reduce communications traffic, changes are only sent according to the specified ***ChangeBefore-Updates*** parameter.

When a VEHICLE starts (or rather first moves into view of the AVMS server), a ***MonitoredVehicleJourney*** instance will be sent; thereafter only changes will be transmitted. Finally a ***VehicleActivityCancellation*** will be transmitted to indicate the journey has completed, been cancelled, or the VEHICLE has moved out of the Vehicle Monitoring zone.

Data is deemed to have changed if at least one of the following pieces of information has changed:

— *Variation from schedule adherence:* the prediction has changed by more than the sensitivity threshold.

— *Completion of journey*: a change remains active until the data assigned to the subscription is explicitly retrieved.

The sensitivity threshold value can be assigned within the subscription. It defines the time span, after which the system regards the change in schedule deviation significant enough to demand communication. With a sensitivity of 2 min, for example, changes in deviations of 2, 4, 6 min, etc. are reported. The change is with respect to the last transmitted arrival and/or departure time.

9.5.3 VehicleMonitoringSubscriptionRequest — Example

The following is an example of a ***VehicleMonitoringSubscriptionRequest***.

```
<SubscriptionRequest>
    <!--=======ENDPOINT REFERENCES================================-->
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!-- Subscription 1 for VIS123 to Place 975 -->
    <VehicleMonitoringSubscriptionRequest>
        <SubscriptionIdentifier>00000456</SubscriptionIdentifier>
```

```
        <InitialTerminationTime>2004-12-17T09:30:47-05:00</InitialTerminationTime>
        <!-- ====== ENDPOINT REFERENCE ====================-->
        <VehicleMonitoringRequest version="1.0">
            <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
            <!--=======TOPIC ==================================== -->
            <VehicleMonitoringRef>VIS123</VehicleMonitoringRef>
            <DestinationRef>Place875</DestinationRef>
            <VehicleMonitoringDetailLevel>minimum</VehicleMonitoringDetailLevel>
        </VehicleMonitoringRequest>
    </VehicleMonitoringSubscriptionRequest>
    <!-- Subscription 2 for VEH222 -->
    <VehicleMonitoringSubscriptionRequest>
        <SubscriptionIdentifier>00000457</SubscriptionIdentifier>
        <InitialTerminationTime>2004-12-17T09:30:47-05:00</InitialTerminationTime>
        <VehicleMonitoringRequest version="1.0">
            <!--====== ENDPOINT REFERENCE ======================-->
            <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
            <!--======= TOPIC ==================================== -->
            <VehicleRef>VEH222</VehicleRef>
            <VehicleMonitoringDetailLevel>calls</VehicleMonitoringDetailLevel>
        </VehicleMonitoringRequest>
        <IncrementalUpdates>false</IncrementalUpdates>
        <ChangeBeforeUpdates>PT2M</ChangeBeforeUpdates>
    </VehicleMonitoringSubscriptionRequest>
</SubscriptionRequest>
```

9.6 VehicleMonitoringDelivery

9.6.1 Introduction

The ***VehicleMonitoringDelivery*** returns the position of a VEHICLE or group of VEHICLEs.

9.6.2 ServiceDelivery with a VehicleMonitoringDelivery — Element

One or more ***VehicleMonitoringDelivery*** elements may be returned as part of a SIRI ***ServiceDelivery***, with a common ***ResponseTimestamp***.

Table 68 — ServiceDelivery / VehicleMonitoringDelivery — Attributes

ServiceDelivery			*+Structure*	See SIRI Part 2 ***ServiceDelivery***
HEADER	*:::*	**1:1**	See ServiceDelivery	
Payload	***VehicleMonitoringDelivery***	**1:***	*+Structure*	See *VehicleMonitoringDelivery* element.

9.6.3 VehicleMonitoringDelivery — Element

A ***VehicleMonitoringDelivery*** is made up of zero, one or many ***VehicleActivity*** elements, each representing a moving VEHICLE following a MONITORED VEHICLE JOURNEY, and indicating its progress relative to the operational schedule. The level of detail included for each ***VehicleActivity*** element may vary by implementation and by request.

Each ***VehicleActivity*** included in the response has its own identifier, issued by the producer: this can be used to reference previously issued ***VehicleActivity*** instances when sending incremental updates – for example the ***VehicleActivityCancellation*** indicates a previous ***VehicleActivity*** element has been removed.

The ***Note*** element allows one or more arbitrary text strings to be associated with the whole delivery, an individual VEHICLE JOURNEY or an individual CALL.

Table 69 — VehicleMonitoringDelivery — Attributes

VehicleMonitoringDelivery			*+Structure*	Describes the progress of a one or more of VEHICLEs along its route.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Vehicle Monitoring Service. Fixed, e.g. '1.1a'.
LEADER	***:::***	**1:1**	xxx*ServiceDelivery*	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***VehicleActivity***	0:*	*+Structure*	Describes the progress of a VEHICLE along its route. *DetailLevel*: minimum.
	VehicleActivity-Cancellation	0:*	*+Structure*	Reference to a previously communicated ***VehicleActivity*** which should not now be used. *DetailLevel*: minimum.
	VehicleActivity Note	0:*	*NLString*	General Text Note associated with delivery. *DetailLevel*: basic.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

9.6.4 VehicleActivity — Element

9.6.4.1 General

Each ***VehicleActivity*** describes the position and relative progress a VEHICLE making a monitored VEHICLE JOURNEY, including scheduled and/or predicted real-time times (see Table 70 below).

Table 70 — VehicleActivity — Attributes

VehicleActivity			*+Structure*	Describes the progress of a single VEHICLE along its route.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which VEHICLE data was recorded.
Currency	***ValidUntilTime***	**1:1**	*xsd:dateTime*	Time until which data is valid.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used for server side cleardown of previous Item instances.
	VehicleMonitoring-Ref	0:1	*VehicleMonitoring-Identifier*	Reference to a Vehicle Monitoring scope.
	MonitoringName	0:*	*NLString*	Name to use to describe monitor. May be included to improve usability of SIRI LITE services. (+SIRI v2.0) One per language.
Stop-Progress-Info	***ProgressBetween-Stops***	0:1	*LocationStructure*	Provides information about the progress of the VEHICLE along its current SERVICE LINK. *DetailLevel*: normal.
	LinkDistance	0:1	*xsd:decimal*	The total distance in metres between the previous stop and the next stop. *DetailLevel*: normal.
	Percentage	0.1	*xsd:decimal*	Percentage along link that VEHICLE has travelled. *DetailLevel*: normal.
Journey-Info	***Monitored-VehicleJourney***	**1:1**	*MonitoredVehicle-Journey Structure*	Provides real-time information about the MONITORED VEHICLE JOURNEY which this VEHICLE is running.
Message	***VehicleActivity-Note***	0:*	*NLString*	Note associated with ***VehicleActivity***. *DetailLevel*: normal.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

9.6.4.2 VehicleActivity / MonitoredVehicleJourney — Element

The ***MonitoredVehicleJourney*** has the same structure as for ***MonitoredStopVisit*** / ***Monitored-VehicleJourney***: whereas for a Stop Visit all CALLs are relative to the stop, for a ***VehicleActivity***, all CALLs are relative to the VEHICLE's current position. Some elements will not be populated in the normal level of detail. Table 71 below shows the elements that will normally be returned for a ***VehicleActivity***.

Table 71 — VehicleActivity / MonitoredVehicleJourney — Attributes

MonitoredVehicleJourney		*DetailLevel*: minimum.
VehicleJourney-Identity	***LineRef***	*DetailLevel*: minimum.
	DirectionRef	*DetailLevel*: minimum.
	FramedVehicle-JourneyRef	See SIRI Part 2 *DetailLevel*: *minimum.*
JourneyPatternInfo	***JourneyPatternRef***	*DetailLevel*: full.
	VehicleMode	*DetailLevel*: minimum.
	RouteRef	*DetailLevel*: full.
	PublishedLineName	*DetailLevel*: minimum.
	GroupOfLinesRef	*DetailLevel*: normal. SIRI 2.0
	DirectionName	*DetailLevel*: basic.
	ExternalLineRef	*DetailLevel*: basic.
VehicleJourneyInfo	***:::***	See SIRI Part 2 *VehicleJourneyInfoGroup*. *DetailLevel*: normal.
DisruptionGroup	***:::***	See SIRI Part 2 *DisruptionGroup*. *DetailLevel*: normal.
JourneyProgressInfo	***:::***	See SIRI Part 2 *JourneyProgressGroup*. *DetailLevel*: normal.
TrainOperationalInfo	***:::***	*DetailLevel*: full. SIRI 2.0
OperationalInfoGroup	***:::***	*DetailLevel*: normal. See SIRI Part 2 ***OperationalInfoGroup***. ***BlockRef*** & ***CourseOfJourneyRef:*** *DetailLevel*: normal. ***VehicleRef:*** *DetailLevel*: basic. ***DriverRef*** & ***DriverName:*** *DetailLevel*: normal. ***TrainNumberRef & JourneyPartInfo*** *DetailLevel*: full.
CallingPattern	***PreviousCalls***	*DetailLevel*: calls, Maximum number of previous > 0.
	PreviousCall	See *PreviousCall* element. *DetailLevel*: minimum.
	MonitoredCall	*DetailLevel*: calls.
	OnwardCalls	*DetailLevel*: calls, Maximum number of previous > 0.
	OnwardCall	*DetailLevel*: calls.
	IsComplete-StopSequence	*DetailLevel*: calls.

9.6.4.3 MonitoredVehicleJourney / PreviousCall — Element

The ***PreviousCall*** element describes a CALL which has already been made in the ***MonitoredVehicle-Journey***. It has the same structure as for ***MonitoredStopVisit*** / ***MonitoredVehicleJourney*** / ***PreviousCall***. It will only be populated for a ***VehicleActivity*** if a ***VehicleMonitoringDetailLevel*** of '*calls*' or '*full'* was requested (see Table 72 below).

Table 72 — VehicleActivity / MonitoredVehicleJourney / PreviousCall — Attributes

PreviousCall		*DetailLevel*: full
Stop Identity	***StopPointRef***	*DetailLevel*: full
	VisitNumber	*DetailLevel*: full
	Order	*DetailLevel*: full
	StopPointName	*DetailLevel*: full
Realtime	***VehicleAtStop***	*DetailLevel*: full
Times	***AimedArrivalTime***	*DetailLevel*: full
	ActualArrivalTime	*DetailLevel*: full
	ExpectedArrivalTime	*DetailLevel*: full
	AimedDepartureTime	*DetailLevel*: full
	ActualDepartureTime	*DetailLevel*: full
	ExpectedDepartureTime	*DetailLevel*: full

9.6.4.4 MonitoredVehicleJourney / MonitoredCall — Element

The ***MonitoredCall*** element describes the CALL by the VEHICLE to the most recently visited stop in the ***MonitoredVehicleJourney***. It has the same structure as for ***MonitoredStopVisit*** / ***MonitoredVehicleJourney*** / ***MonitoredCall***. It will only be populated for a ***VehicleActivity*** if a ***VehicleMonitoringDetailLevel*** of '*calls'* or '*full'* was requested (see Table 73 below).

Table 73 — VehicleActivity / MonitoredVehicleJourney /MonitoredCall — Attributes

MonitoredCall		*DetailLevel*: normal
Stop Identity	***StopPointRef***	*DetailLevel*: normal
	VisitNumber	*DetailLevel*: normal
	Order	*DetailLevel*: full
	StopPointName	*DetailLevel*: normal
Real-time	***VehicleAtStop***	*DetailLevel*: normal
	VehicleLocationAtStop	*DetailLevel*: full
	ReversesAtStop	*DetailLevel*: normal
Stop Info	***TimingPoint***	*DetailLevel*: full
	BoardingStretch	*DetailLevel*: full
	RequestStop	*DetailLevel*: full
	OriginDisplay	*DetailLevel*: full
	DestinationDisplay	*DetailLevel*: full
Call Info	***CallNote***	*DetailLevel*: full
	SituationRef	*DetailLevel*: normal
	FacilityChange	*DetailLevel:* normal
Arrival	***AimedArrivalTime***	*DetailLevel*: calls
	ActualArrivalTime	*DetailLevel*: calls
	ExpectedArrivalTime	*DetailLevel*: calls
	LatestExpectedArrivalTime	*DetailLevel*: calls. SIRI 2.0

	ExpectedArrival-PredictionQuality	*DetailLevel*: full SIRI 2.0
	ArrivalStatus	*DetailLevel*: calls
	ArrivalPlatformName	*DetailLevel*: full
	ArrivalProximityText	*DetailLevel*: full SIRI 2.0
	ArrivalBoardingActivity	*DetailLevel*: full
	ArrivalStopAssignment	*DetailLevel*: calls SIRI 2.0
	ArrivalOperatorRefs	*DetailLevel*: full SIRI 2.0
Departure	***AimedDepartureTime***	*DetailLevel*: calls
	ActualDepartureTime	*DetailLevel*: calls
	ExpectedDepartureTime	*DetailLevel*: calls
	ProvisionalExpected-DepartureTime	*DetailLevel*: calls SIRI 2.0
	EarliestExpected-DepartureTime	*DetailLevel*: calls SIRI 2.0
	ExpectedDeparture-PredictionQuality	*DetailLevel*: full SIRI 2.0
	AimedLatestPassenger-AccessTime	*DetailLevel*: calls SIRI 2.0
	Expected LatestPassenger-AccessTime	*DetailLevel*: calls SIRI 2.0
	DepartureStatus	*DetailLevel*: calls
	DepartureProximityText	*DetailLevel*: full SIRI 2.0
	DeparturePlatformName	*DetailLevel*: full
	DepartureBoardingActivity	*DetailLevel*: full
	DepartureStopAssignment	*DetailLevel*: calls SIRI 2.0
	DepartureOperatorRefs	*DetailLevel*: full SIRI 2.0
Interval	***AimedHeadwayInterval***	*DetailLevel*: calls
	ExpectedHeadwayInterval	*DetailLevel*: calls
Distance	***DistanceFromStop***	*DetailLevel*: calls SIRI 2.0
	NumberOfStopsAway	*DetailLevel*: calls SIRI 2.0
any	***Extensions***	Placeholder for user extensions

9.6.4.5 MonitoredVehicleJourney / OnwardCall — Element

The ***OnwardCall*** element describes a CALL which has still to be made in the ***MonitoredVehicleJourney***. It has the same structure as for ***MonitoredStopVisit*** / ***MonitoredVehicleJourney*** / ***OnwardCall***.

It will only be populated for a ***VehicleActivity*** / ***MonitoredVehicleJourney*** if a ***VehicleMonitoring-DetailLevel*** of '*calls*' is requested. The number of onward CALLs to include is given by the ***Maximum-NumberOfCalls*** / ***Onward*** element.

Table 74 — VehicleActivity / MonitoredVehicleJourney / OnwardCall — Attributes

OnwardCall		*DetailLevel*: calls
StopIdentity	***StopPointRef***	*DetailLevel*: calls
	VisitNumber	*DetailLevel*: calls
	Order	*DetailLevel*: full
	StopPointName	*DetailLevel*: calls
Progress	***VehicleAtStop***	*DetailLevel*: calls
CallTimes	***TimingPoint***	*DetailLevel*: full
	AimedArrivalTime	*DetailLevel*: full
	ExpectedArrivalTime	*DetailLevel*: calls
	ExpectedArrival-PredictionQuality	*DetailLevel*: full SIRI 2.0
	ArrivalStatus	*DetailLevel*: full
	ArrivalPlatformName	*DetailLevel*: full
	ArrivalProximityText	*DetailLevel*: full SIRI 2.0
	ArrivalBoardingActivity	*DetailLevel*: full
	ArrivalStopAssignment	*DetailLevel*: full SIRI 2.0
	ArrivalOperatorRefs	*DetailLevel*: full SIRI 2.0
	AimedDepartureTime	*DetailLevel*: full
	ExpectedDepartureTime	*DetailLevel*: calls
	ProvisionalExpected-DepartureTime	*DetailLevel*: full SIRI 2.0
	EarliestExpected-DepartureTime	*DetailLevel*: full SIRI 2.0
	ExpectedDeparture-PredictionQuality	*DetailLevel*: full SIRI 2.0
	AimedLatestPassenger-AccessTime	*DetailLevel*: full SIRI 2.0
	Expected LatestPassenger-AccessTime	*DetailLevel*: full SIRI 2.0
	DepartureStatus	*DetailLevel*: full
	DepartureProximityText	*DetailLevel*: full SIRI 2.0
	DeparturePlatformName	*DetailLevel*: full
	DepartureBoardingActivity	*DetailLevel*: full
	DepartureStopAssignment	*DetailLevel*: full SIRI 2.0
	DepartureOperatorRefs	*DetailLevel*: full SIRI 2.0
	AimedHeadWayInterval	*DetailLevel*: full
	EstimatedHeadwayInterval	*DetailLevel*: full
CallStatus	***ArrivalStatus***	*DetailLevel*: full
	DepartureStatus	*DetailLevel*: full
Merge cells	***DistanceFromStop***	*DetailLevel*: full SIRI 2.0
	NumberOfStopsAway	*DetailLevel*: full SIRI 2.0

9.6.5 VehicleActivityCancellation — Element

The ***VehicleActivityCancellation*** element is used to cleardown an earlier ***VehicleActivity***, i.e. indicate that it is no longer current, for example because the journey has completed, or the VEHICLE has moved out of the monitored zone. This is useful in particular if only updates are being exchanged (see Table 75 below).

A ***VehicleActivityCancellation*** shall contain sufficient information to identify an earlier ***VehicleActivity*** entry; either an explicit reference, or reference values sufficient to uniquely distinguish the cleared ***VehicleActivity*** from other instances.

Table 75 — VehicleActivityCancellation — Attributes

VehicleActivityCancellation			*+Structure*	Cancellation of a previously communicated ***VehicleActivity.*** The ***VehicleActivity*** should now be removed from the system.
Endpoint	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which data was recorded.
Event-Identity	***ItemRef***	0:1	*ItemIdentifier*	Reference to a previous ***VehicleActivity*** which this item cancels.
	Vehicle-MonitoringRef	0:1	*→VehicleMonitoring-Code*	Reference to a Vehicle Monitoring Scope. (LOGICAL DISPLAY)
	Vehicle-JourneyRef	0:1	*+Structure*	Frame qualified reference to a DATED VEHICLE JOURNEY that is being deleted. See SIRI Part 2.
	DataFrame Ref	0:1	*→VehicleMonitoring-Code*	Reference to a Data frame within which dated reference is unique.
	Dated-Vehicle-JourneyRef	**1:1**	*→VehicleJourney-Code*	Reference to a DATED VEHICLE JOURNEY that is being deleted.
	LineRef	0:1	*→LineCode*	Reference to a LINE of journey that is being deleted.
	DirectionRef	0:1	*→DirectionCode*	Reference to a DIRECTION of journey that is being deleted.
Journey-PatternInfo	**:::**	0:1	*JourneyPatternInfo-Group*	See SIRI Part 2 *JourneyPatternInfoGroup*.
Message	***Reason***	0:*	*NLString*	Message associated with delivery. One per language (Unbounded 0:* since +SIRI v2.0).
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

9.6.6 VehicleMonitoringDelivery — Examples

9.6.6.1 VehicleMonitoringDelivery — Basic Example

The following is an example of a ***VehicleMonitoringDelivery***. It shows a single VEHICLE, for a detail level of ***Normal***.

```
<ServiceDelivery>
    <!--========ENDPOINT ================================== -->
    <RequestorRef>NADER</RequestorRef>
    <RequestRef>2004-12-17T09:30:47-05:00</RequestRef>
    <!--========PAYLOAD ================================== -->
    <!--========FIRST SUBSCRIPTION========================= -->
    <VehicleMonitoringDelivery version="1.0">
        < ResponseTimestamp>2004-12-17T09:30:47-05:00</ ResponseTimestamp>
        <SubscriberRef> NADER </SubscriberRef>
```

```
        <SubscriptionRef>00047</SubscriptionRef>
        <VehicleActivity>
            <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
            <ValidUntilTime>2004-12-17T09:30:47-05:00</ValidUntilTime>
            <ItemIdentifier >915468</ItemIdentifier >
            <VehicleMonitoringRef>MYACACT</VehicleMonitoringRef>
            <ProgressBetweenStops>
                <LinkDistance>3.14</LinkDistance>
                <Percentage>60.5</Percentage>
            </ProgressBetweenStops>
            <!-- MONITORED VEHICLE JOURNEY -->
            <MonitoredVehicleJourney>
                <!-- LINE GROUP -->
                <LineRef>Line123</LineRef>
                <!-- JOURNEY IDENTITY GROUP -->
                <FramedVehicleJourneyRef>
                    <DataFrameRef>2004-12-17</DataFrameRef>
                    <DatedVehicleJourneyRef>167567</DatedVehicleJourneyRef>
                </FramedVehicleJourneyRef>
                <!--JOURNEY PROGRESS GROUP -->
                <Monitored>true</Monitored>
                <VehicleLocation>
                    <Longitude>180</Longitude>
                    <Latitude>90</Latitude>
                </VehicleLocation>
                <Delay>PT2M</Delay>
                <VehicleRef>VEH987654</VehicleRef>
                <OnwardCalls>
                    <OnwardCall>
                        <StopPointRef>HLTST012</StopPointRef>
                        <StopPointName>Church</StopPointName>
                    </OnwardCall>
                </OnwardCalls>
            </MonitoredVehicleJourney>
        </VehicleActivity>
    </VehicleMonitoringDelivery>
</ServiceDelivery>
```

9.6.6.2 VehicleMonitoringDelivery — Extended Example

The following is a more verbose example of a ***VehicleMonitoringDelivery***, populated with more detail including previous and onward CALL details.

```
<ServiceDelivery>
    <!--========ENDPOINT ==================================== -->
    <RequestorRef>NADER</RequestorRef>
    <RequestRef>2004-12-17T09:30:47-05:00</RequestRef>
    <!--========PAYLOAD ==================================== -->
    <VehicleMonitoringDelivery version="0.1d">
        <ResponseTimestamp>2004-12-17T09:30:47-05:00</ResponseTimestamp>
        <SubscriberRef>NADER</SubscriberRef>
        <SubscriptionRef>00047</SubscriptionRef>
        <Status>true</Status>
        <ValidUntil>2004-12-17T09:30:47-05:00</ValidUntil>
        <ShortestPossibleCycle>P1Y2M3DT10H30M</ShortestPossibleCycle>
        <VehicleActivity>
            <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
            <ValidUntilTime>2004-12-17T09:30:47-05:00</ValidUntilTime>
            <ItemIdentifier >EV000123</ItemIdentifier >
            <VehicleMonitoringRef>ACT019456</VehicleMonitoringRef>
            <ProgressBetweenStops>
                <LinkDistance>3.14</LinkDistance>
```

```
            <Percentage>60.5</Percentage>
        </ProgressBetweenStops>
        <!-- MONITORED VEHICLE JOURNEY -->
        <MonitoredVehicleJourney>
            <!-- JOURNEY IDENTITY GROUP -->
            <LineRef>Line123</LineRef>
            <DirectionRef>OUT</DirectionRef>
            <JourneyRef>
                <DataFrameRef>2004-12-17</DataFrameRef>
                <DatedVehicleJourneyRef>6746</DatedVehicleJourneyRef>
            </JourneyRef>
            <!-- LINE GROUP -->
            <PublishedLineName>123</PublishedLineName>
            <Destination>
                <DestinationRef>PLACE45</DestinationRef>
                <DestinationName>Paradise Park</DestinationName>
            </Destination>
            <!-- SERVICE INFO GROUP -->
            <OperatorRef>OP22</OperatorRef>
            <ProductCategoryRef>PDCATEXPRESS</ProductCategoryRef>
            <ServiceFeatureRef>SERVCCAT551</ServiceFeatureRef>
            <JourneyNote>Kensall Green</JourneyNote>
            <!--JOURNEY PROGRESS GROUP -->
            <Monitored>true</Monitored>
            <InCongestion>false</InCongestion>
            <VehicleLocation>
                <Longitude>180</Longitude>
                <Latitude>90</Latitude>
            </VehicleLocation>
            <Delay>PT2M</Delay>
            <ProgressStatus>On time</ProgressStatus>
            <!-- Operational Info GROUP -->
            <TrainBlockPart>
                <NumberOfBlockParts>1</NumberOfBlockParts>
                <TrainPartRef>3456</TrainPartRef>
                <PositionOfTrainBlockPart>1</PositionOfTrainBlockPart>
            </TrainBlockPart>
            <BlockRef>BLOCK765</BlockRef>
            <CourseOfJourneyRef>RUN765</CourseOfJourneyRef>
            <VehicleRef>VEH987654</VehicleRef>
            <!-- MONITORED CALLING PATTERN GROUP -->
            <PreviousCalls>
                <PreviousCall>
                    <StopPointRef>HLT0011</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>HLTST012</StopPointRef>
                    <VisitNumber>4</VisitNumber>
                    <StopPointName>String</StopPointName>
                    <VehicleAtStop>false</VehicleAtStop>
                    <AimedArrivalTime>2004-12-17T09:30:56-05:00</AimedArrivalTime>
                    <AimedDepartureTime>2004-12-17T09:30:57-05:00</AimedDepartureTime>
                    <ExpectedArrivalTime>2004-12-17T09:30:56-05:00</ExpectedArrivalTime>
                    <ExpectedDepartureTime>2004-12-17T09:30:57-05:00</ExpectedDepartureTime>
                </OnwardCall>
            </OnwardCalls>
        </MonitoredVehicleJourney>
        <Message>hello vehicle</Message>
    </VehicleActivity>
    <VehicleActivityCancellation>
        <RecordedAtTime>2004-12-17T09:30:47-05:00</RecordedAtTime>
```

```
            <VehicleMonitoringRef>9876542</VehicleMonitoringRef>
            <FramedVehicleJourneyRef>
                <DataFrameRef>2001-12-17</DataFrameRef>
                <DatedVehicleJourneyRef>09867</DatedVehicleJourneyRef>
            </FramedVehicleJourneyRef>
                <LineRef>Line123</LineRef>
                <DestinationRef>HELL</DestinationRef>
            <Reason>Gone home</Reason>
        </VehicleActivityCancellation>
        <Message>hello delivery</Message>
    </VehicleMonitoringDelivery>
</ServiceDelivery>
```

10 Connection Timetable Data Service [CT]

10.1 Purpose

The SIRI Connection Timetable Service is used for the exchange of schedule data for potential feeder VEHICLE JOURNEYs to a connection zone. It is used in conjunction with the SIRI Connection Monitoring Service. The service is location-related, i.e. all requests and replies relate to specific CONNECTION links, as identified by CONNECTION link identifiers.

The exchange of planned schedule data is only relevant for timetable i.e. ***DatedVehicleJourney***-based 'connection protection', which requires advance knowledge of the feeder VEHICLE JOURNEYs. This can be achieved through a common schedule supply to both systems, made outside of SIRI, or through the dynamic exchange of data using the SIRI Connection Timetable service.

The SIRI Connection Timetable service is initiated by the distributor, that is, the AVMS managing the distributor journey. The distributor requests the VEHICLE JOURNEYs at a defined CONNECTION link from the feeder, that is, the AVMS running the feeder system. Depending on the implementation capabilities and data coverage, the request may be restricted to foreign LINEs and DIRECTIONs only. Whether foreign journeys are included is given by a capability.

The feeder satisfies the request by returning a table of feeder arrivals to the desired CONNECTION link. These can be computed from the ***TargetedInterchange*** children of the ***DatedCall*** in the Production timetable. The journey identifier supplied for every DATED VEHICLE JOURNEY can be used subsequently in requests for the SIRI Connection Monitoring service to reference the connecting journeys.

The data provided by the Connection Timetable service can be used for two purposes:

— To subscribe to specific known VEHICLE JOURNEYs.

— To determine possible connections between journeys (i.e. SERVICE JOURNEY INTERCHANGEs) in advance so that the distributor system is aware of those journeys from which passengers may wish to transfer.

The SIRI Connection Timetable Service comprises the ***ConnectionTimetableRequest*** message used to specify the contents of request or subscription messages, and the ***ConnectionTimetableDelivery*** message used to deliver the response. The related ***ConnectionTimetableSubscriptionRequest*** message allows a subscriber to request asynchronous updates for the service: it contains an embedded ***ConnectionTimetable-Request,*** along with further parameters controlling the asynchronous delivery.

The ***ConnectionTimetableRequest*** has topic parameters to filter the information by journey, LINE, and DIRECTION, etc, and policy parameters to control the way any parameterised aspects processing should be undertaken.

The ***ConnectionTimetableDelivery*** returns information about one or more SERVICE JOURNEY INTERCHANGEs, each as a ***TimetabledFeederArrival*** instance, which may be populated to different levels

of detail depending on the application requirements. The same response message is also used by the SIRI Connection Monitoring Service.

10.2 Reference Data

10.2.1 Content referencing

Implementation of full connection protection requires agreement of data for CONNECTION Links, LINEs, DIRECTIONs and Product Category Codes.

Transmodel by convention uses the terms CONNECTION link for the physical connection of two stops between which transfers take place, and INTERCHANGE for the corresponding planned meeting between two services that takes place at the CONNECTION: thus a CONNECTION link connects two SCHEDULED STOP POINTs (which may be the same physical stop). The SCHEDULED STOP POINTs and CONNECTION link may lie within a CONNECTION link.

SIRI allows for identifiers to be assigned both to the CONNECTION link, and to the INTERCHANGE. The CONNECTION link is treated as a simple point-to-point identifier: this may be projected on to a more complex path model with mobility hazards etc, by client systems. The simplest case of exchanging reference data for connection protection requires only the supply of common CONNECTION link codes (***ConnectionLinkRef***). An allocation table is created, assigning the abstract location codes (Connection Links) to the internal location codes (stops or areas). On the side of the distributor system it is also necessary to provide the changeover time.

This type of definition could look as follows:

For operator A (feeder AVMS A):

Table 76 — Example of Simple Connection Data - Feeder

ConnectionLinkRef	*ForeignOperatorCode*	Internal stop code
12345	AVMS B	3642
12346	AVMS B	4564
35678	AVMS B	7765

For operator B (distributor AVMS B):

Table 77 — Example of Simple Connection Data - Distributor

ConnectionLinkRef	*ForeignOperatorCode*	Internal stop code	Changeover time
12345	AVMS A	2345	0
12346	AVMS A	3687	0
35678	AVMS A	7566	1

Agreement on the above is sufficient to make requests for feeder data. However, it is not sufficient to make selections based on specific connection relationships of LINEs, DIRECTIONs or time of day. This requires a more extended data management, defining additional CONNECTION link relationships.

Table 78 — Example of Extended Connection Data

Foreign-Operator-Code	ConnectionLinkRef	LineRef (Feeder)	DirectionRef (Feeder)	LineRef (Distributor)	DirectionRef (Distributor)	Time
AVMS A	12345	10	Zoo	12	CSt.	1000-1200
AVMS A	12345	10	Station	12	CSt.	1000-1400

This example specifies that VEHICLE JOURNEYs in two DIRECTIONs (*Zoo, Station)* of one LINE (*10*) of foreign operator *AVMS A* are potential feeders, if the distributor VEHICLE of the actual operator is travelling on line 12 in DIRECTION *'CSt',* but only between 10:00 a.m. and 12:00 p.m. or 10:00 a.m. and 2:00 p.m. respectively.

From the above table it can be seen that in addition to the CONNECTION link, the foreign LINE or DIRECTION shall also be known to both systems. The local LINE and DIRECTION information needs to be known also to the foreign system, as for return messages or subscriptions, it is necessary to provide distributor information to the feeder.

Data management is always carried out from the point of view of the distributor. On the feeder side, it is not necessary to manage the foreign distributor LINE or DIRECTION.

If used, product types (***ProductCategoryRef***) and Service Features (***ServiceFeatureRef***) shall also be agreed between the operators.

10.2.2 Availability of Data

The end of the operating day of one AVMS does not necessarily coincide with the end of the operating day of another AVMS. In consequence, data may not be available for the entire time frame of requests from one system to another. To ensure the data consumer (distributor) is aware of the horizon end, there is an optional ***ValidUntil*** element defined in the ***ResponseStatus*** element message of the ***ConnectionTimetableSubscriptionResponse*** message. If the feeder receives a request, which goes beyond its data availability horizon, it can return the end of the horizon here. If the ***ValidUntil*** element is missing, the distributor can assume that its request can be satisfied for the required period.

If any part of the request time frame lies outside the data horizon of the producer, the ***ResponseStatus*** should be false and the value returned in ***ValidUntil*** *to* indicate the time window available to satisfy the request.

10.2.3 Updating During the Course of a Journey

When a subscription is set up, the current data set of ***TimetabledFeederArrival*** instances is delivered. If further arrivals take place, further updates are sent to subscribers.

Updates cover additional VEHICLEs only. The reason for the update is not given (for example a reinforcement VEHICLE JOURNEY, special purpose VEHICLE, diversions, etc.)

Subsequent changes to the VEHICLE JOURNEY itself (origin stop, departure time, MODE of transport, product) are not communicated, or are communicated by the SIRI Connection Monitoring service. Vehicle journeys that no longer reach the CONNECTION link as a result of a dispatch action are removed from the data set by an explicit message (***MonitoredFeederArrivalCancellation***).

Replacement VEHICLE JOURNEYs for any that have failed should be transparent (recommendation). It is simply the same VEHICLE JOURNEY with a different VEHICLE.

The production of updates by the Producer is optional. Consumers shall always be able to receive updates.

10.3 Capability and Permission Matrices

10.3.1 Capability Matrix

The following set of required and optional capabilities is defined for the Connection Timetable Service. If the service supports Capability Discovery, the ***ConnectionTimetableCapabilitiesRequest*** / ***ConnectionTimetableCapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 79 — ConnectionTimetableCapabilities Matrix

ConnectionTimetableCapabilities				*+Structure*	Capabilities describing implementation of Connection Timetable service.
inherit	***:::***		**1:1**	*See xxx-Capability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***		0:1	*+Structure*	Which optional filtering features are supported.
		FilterByLineRef	0:1	*xsd:boolean*	Whether results can be filtered by LINE. Required Capability: Default is *'true'*.
		FilterByConnection-LinkRef	0:1	*xsd:boolean*	Whether results can be filtered by CONNECTION link. Default is *'true'*.
Request-Policy	***RequestPolicy***		0:1	*+Structure*	Which optional features of the Request Policy are supported by the service?
		Language	**1:***	*xsd:language*	National languages used by service.
		Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
				choice	Location coordinate system for results.
		a ***GmlCoordinate-Format***	**1:1**	*SrsNameType*	Default coordinate format is given by a GML value.
		b ***WgsDecimal-Degrees***		*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
		ForeignJourneysOnly	0:1	*xsd:boolean*	Whether service returns only foreign journeys. Default is *'false'*.
Sub-scription-Policy	***SubscriptionPolicy***		0:1	*+Structure*	Which features of Subscription Policy are supported by service?
		HasIncremental-Updates	**1:1**	*xsd:boolean*	Whether incremental updates can be specified for updates Default is *'true'*.
		HasChangeSensitivity	0:1	*xsd:boolean*	Whether change threshold can be specified for updates. Default is *'true'*.
Access-Control	***AccessControl***		0:1	*+Structure*	Which optional access control features are supported by service?
		RequestChecking	0:1	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
		CheckOperatorRef	0:1	*xsd:boolean*	If access *control* is supported, whether access control by OPERATOR is supported. Default is *'true'*.
		CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
		CheckConnection-LinkRef	0:1	*xsd:boolean*	If access control is supported, whether access control by CONNECTION is supported. Default is *'true'*.
any	***Extensions***		0:1	xsd:any*	Placeholder for user extensions.

10.3.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***ConnectionTimetableCapabilitiesResponse*** response can include the access permissions for the requestor participant to access data.

Table 80 — Connection Timetable Service — Permissions

ConnectionTimetablePermission			*+Structure*	Permissions to use implementation of Connection Timetable service
In-herit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant. See Part 2.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant. See Part 2.
	ConnectionLinkPermissions	0:1	*+Structure*	CONNECTION link permissions for participant. See Part 2.

10.4 ConnectionTimetableRequest

10.4.1 ConnectionTimetableRequest — Element

ConnectionTimetableRequest can be used both in a direct request and in a subscription. Connection can be requested by a known link identifier and LINE identifier to return all the connections on a LINE or area, or a combination of both. It is also possible to request the movement for a given VEHICLE. The primary ***Topic*** term is the CONNECTION LINK for which information is to be returned.

Table 81 — ConnectionTimetableRequest — Attributes

ConnectionTimetableRequest			*+Structure*	Request for information about messages.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of *Connection Timetable* Service.
Endpoint Properties	***Request-Timestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	MessageIdentifier	0:1	*MessageQualifier*	
Topic	***ArrivalWindow***	0:1	*ClosedTimeInterval*	Time frame for which data is to be supplied. The reference is the arrival time of the feeder at the CONNECTION link. If absent use data horizon of server system.
	StartTime	**1:1**	*xsd:dateTime*	Earliest arrival time of VEHICLE. Inclusive start time.
	EndTime	**1:1**	*xsd:dateTime*	Latest arrival time of VEHICLE. Inclusive end of the time.
	ConnectionLink-Ref	**1:1**	*→Connection-LinkCode*	Reference to the CONNECTION link for which data is to be returned. If absent, all channels.
	LineRef	0:1	*→LineCode*	Filter for the feeder LINE, which is to supply data.
	DirectionRef	0:1	*→DirectionCode*	Filter for the feeder DIRECTION, for which data is to be supplied.
Request Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *NationalLanguage*.
	Include-Translations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

LineRef and ***DirectionRef*** are independent: it is possible to specify a request that only includes a DIRECTION filter without a LINE filter.

The earliest arrival time (***ArrivalWindow*** / ***StartTime***) element shall not be set to a value before the start of the actual data horizon of the server, and the latest arrival time (***ArrivalWindow*** / ***EndTime***) shall not be after the end of the horizon.

10.4.2 ConnectionTimetableRequest — Example

The following is an example of a ***ConnectionTimetableRequest***.

```
<ServiceRequest>
    <!--======ENDPOINT REFERENCES================================-->
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <RequestorRef>NADER</RequestorRef>
    <ConnectionTimetableRequest version="0.1d">
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--========TOPIC ==================================== -->
        <ArrivalWindow>
            <StartTime>2004-12-17T09:30:47-05:00</StartTime>
            <EndTime>2004-12-17T10:30:47-05:00</EndTime>
        </ArrivalWindow>
        <ConnectionLinkRef>EH00001</ConnectionLinkRef>
        <LineRef>LINE77</LineRef>
    </ConnectionTimetableRequest>
</ServiceRequest>
```

10.5 ConnectionTimetableSubscriptionRequest

10.5.1 ConnectionTimetableSubscriptionRequest — Element

The ***ConnectionTimetableSubscriptionRequest*** (see Table 82 below) requests the asynchronous delivery of the information described by a ***ConnectionTimetableRequest***.

Table 82 — ConnectionTimetableSubscriptionRequest — Attributes

ConnectionTimetableSubscriptionRequest			*+Structure*	Request for a subscription to the Connection Timetable Service.
Identity	***SubscriberRef***	0:1	*→Participant-Code*	See SIRI Part 2 Common ***SubscriptionRequest*** parameters.
	SubscriptionIdentifier	**1:1**	*Subscription-Qualifier*	
Lease	***InitialTerminationTime***	**1:1**	*xsd:dateTime*	
Request	***Connection-TimetableRequest***	**1:1**	*+Structure*	See ***ConnectionTimetableRequest***.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

The ***InitialTerminationTime*** should be equal to or later than the latest arrival time on the request.

10.5.2 ConnectionTimetableSubscriptionRequest — Example

The following is an example of a ***ConnectionTimetableSubscriptionRequest***.

```
<SubscriptionRequest>
    <!--======ENDPOINT REFERENCES================================-->
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <!-- Subscription 1 for VIS123 to Place 975 -->
    <ConnectionTimetableSubscriptionRequest>
        <SubscriptionIdentifier>00000456</SubscriptionIdentifier>
```

```
        <InitialTerminationTime>2004-12-17T09:30:47-05:00</InitialTerminationTime>
        <!-- ======ENDPOINT REFERENCE================================--->
        <ConnectionTimetableRequest version="1.0">
            <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
            <!--=======TOPIC ==================================== -->
            <ConnectionLinkRef>I23</ ConnectionLinkRef>
        </ConnectionTimetableRequest>
    </ ConnectionTimetableSubscriptionRequest>
</SubscriptionRequest>
```

10.6 ConnectionTimetableDelivery

10.6.1 Introduction

The ***ConnectionTimetableDelivery*** returns the feeder arrivals to a specified link in response to a ***ConnectionTimetableRequest***.

To satisfy the request, the feeder system that receives the request searches its own planning data (including current dispatch actions) and compiles the departure tables of ***FeederArrival*** Instances.

10.6.2 ServiceDelivery with a ConnectionTimetableDelivery — Element

One or more ***ConnectionTimetableDelivery*** elements may be returned as part of a SIRI ***ServiceDelivery***, with a common ***ResponseTimestamp*** (see Table 83 below).

Table 83 — ServiceDelivery / ConnectionTimetableDelivery — Attributes

ServiceDelivery			+*Structure*	See SIRI Part 2 ***ServiceDelivery***
HEADER	:::	**1:1**	See ServiceDelivery	
Payload	***ConnectionTimetable-Delivery***	**1:***	+*Structure*	See *ConnectionTimetableDelivery* element.

10.6.3 ConnectionTimetableDelivery — Element

10.6.3.1 Introduction

A ***ConnectionTimetableDelivery*** is made up of zero, one or many ***TimetabledFeederArrival*** elements, each representing a distinct planned SERVICE JOURNEY INTERCHANGE between VEHICLE JOURNEYs over which passengers may transfer (see Table 84 below).

Table 84 — ConnectionTimetableDelivery — Attributes

ConnectionTimetableDelivery			+*Structure*	Delivery of Feeder SERVICE JOURNEY INTERCHANGEs for Connection Timetable Service.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Connection Timetable Service. Fixed e.g., '1.1a'.
LEADER	***:::***	**1:1**	xxx***Service-Delivery***	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***TimetabledFeeder-Arrival***	0:*	+*Structure*	Planned arrival of a feeder VEHICLE JOURNEY at a CONNECTION link.
	TimetabledFeeder-ArrivalCancellation	0:*	+*Structure*	A deletion of a managed feeder SERVICE JOURNEY INTERCHANGE at a CONNECTION link. See ***TimetabledFeederArrival*** element.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

10.6.3.2 TimetabledFeederArrival — Element

Each ***TimetabledFeederArrival*** describes an inbound connection, together making up a connection timetable for the CONNECTION LINK (see Table 85 below).

Table 85 — TimetabledFeederArrival — *Attributes*

TimetabledFeederArrival			*+Structure*	Planned SERVICE JOURNEY INTERCHANGE from a feeder VEHICLE JOURNEY at a CONNECTION link.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which ***FeederArrival*** was recorded.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used for server side cleardown of previous ***TimetabledFeederArrival*** instances.
Interchange Identity	***InterchangeRef***	0:1	*→Interchange-Code*	Reference to the SERVICE JOURNEY INTERCHANGE between two journeys for which data is being returned.
	ConnectionLink-Ref	**1:1**	*→Connection-LinkCode*	Reference to a CONNECTION link or ACCESS zone for which data is to be returned. Associated with known feeder arrival and distributor departure SCHEDULED STOP POINTS.
	StopPointRef	0:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT. Usually the same as that for the CONNECTION link for which data is to be returned.
	VisitNumber	0:1	*VisitNumber-Type*	Order of visit to a stop within a JOURNEY PATTERN.
	Order	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Feeder stop name of CONNECTION link. One per language (Unbounded 0:* since +SIRI v2.0).
Journey Info	***FeederJourney***	**1:1**	*InterchangeJourneyStructure*	Provides information about the MONITORED VEHICLE JOURNEY which this VEHICLE is running. See ***FeederJourney*** element.
Call time	***Aimed-ArrivalTime***	**1:1**	*xsd:dateTime*	Planned arrival time at the feeder stop of CONNECTION link.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions

10.6.3.3 FeederJourney — Element

Each ***FeederJourney*** describes information about the connecting journey (see table 81 below).

Optional information in the ***FeederJourney*** element other than ***JourneyRef, LineRef***, and ***DirectionRef*** is not required in order to carry out connection protection. It can be provided for informative purposes, for example to give extra information to the dispatcher, which may be useful in the course of telephone conversations with other control centres or for logging purposes.

Table 86 — FeederJourney — Attributes

FeederJourney			*InterchangeJourneyStructure*	Planned SERVICE JOURNEY INTERCHANGE from a feeder VEHICLE JOURNEY at a CONNECTION link.
Vehicle-Journey-Identity	***LineRef***	0:1	*→LineCode*	Identifier for the LINE of the feeder journey.
	DirectionRef	0:1	*→DirectionCode*	Reference to the DIRECTION the VEHICLE is running along the LINE, for example, "in" or "out", “clockwise”. Distinct from a destination.
	FramedVehicle-JourneyRef	0:1	*+Structure*	A frame qualified reference to the DATED VEHICLE JOURNEY that the VEHICLE is making, unique within data horizon of producer. See SIRI Part 2.
Journey-PatternInfo	***:::***	0:1	*JourneyPattern-InfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Vehicle-JourneyInfo	***:::***	0:1	*VehicleJourney-InfoGroup*	See SIRI Part 2 ***VehicleJourneyInfoGroup***.
Disruption-Group	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***DisruptionGroup***.
Operational Info	***:::***	0:1	*OperationalInfo-Group*	See SIRI Part 2 ***OperationalInfoGroup.***
Progress	***Monitored***	0:1	*xsd:boolean*	Whether there is real-time information available for journey, if not present, not known.
Times	***AimedDeparture-Time***	0:1	*xsd:dateTime*	On a Feeder journey a Timetabled arrival time of the VEHICLE JOURNEY at the CONNECTION link.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

10.6.3.4 TimetabledFeederArrivalCancellation — Element

Each ***TimetabledFeederArrivalCancellation*** describes the cancellation of a planned feeder connection.

The text elements (LINE, DIRECTION) are for informative purposes only for the dispatcher, as the VEHICLE JOURNEY is uniquely referenced via the ***FeederVehicleJourneyRef***.

Table 87 — TimetabledFeederArrivalCancellation — Attributes

TimetabledFeederArrival-Cancellation			*+Structure*	Cancellation of previous SERVICE JOURNEY INTERCHANGE as specified by a ***TimetabledFeederArrival***.
Log	***Recorded-AtTime***	**1:1**	*xsd:dateTime*	Time at which ***FeederArrivalCancellation*** was recorded.
Identity	***ItemRef***	0:1	*ItemIdentifier*	Reference to a previous ***FeederArrival*** which this item cancels.
Interchange Identity	***Interchange-Ref***	0:1	*→InterchangeCode*	Reference to the SERVICE JOURNEY INTERCHANGE between two journeys for which data is being returned.
	Connection-LinkRef	**1:1**	*→Connection-LinkCode*	Reference to the CONNECTION link for which data is to be returned.
	StopPointRef	0:1	*→StopPointCode*	Reference to a SCHEDULED STOP POINT. Usually this will be the same as on the CONNECTION link for which data is to be returned.

	VisitNumber	0:1	*StopSequenceType*	Order of visit to a given stop within a JOURNEY PATTERN.
	Order	0:1	*StopSequenceType*	For implementations for which the overall Order is not used for VisitNumber, (i.e. if VisitNumberIsOrder is false) then can be used to associate the stop Order as well if useful.
	Stop-PointName	0:*	*NLString*	Feeder stop name of CONNECTION Link. One per language (Unbounded 0:* since +SIRI v2.0).
JourneyInfo	***LineRef***	**1:1**	*→LineCode*	Reference to a LINE.
	DirectionRef	**1:1**	*→DestinationCode*	Reference to a DIRECTION.
	VehicleJourneyRef	**1:1**	*FramedVehicle-JourneyRefStructure*	A reference the feeder VEHICLE JOURNEYs. See SIRI Part 2.
Journey-Pattern Info	**:::**	0:1	*JourneyPatternInfo-Group*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Info	***Reason***	0:*	*NLString*	Reason for failure of SERVICE JOURNEY INTERCHANGE. One per language (Unbounded 0:* since +SIRI v2.0).
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

10.6.4 ConnectionTimetableDelivery — Example

The following is an example of a ***ConnectionTimetableDelivery***.

```
<ServiceDelivery>
    <!--=======HEADER==================================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <!--=======Functional SERVICE PAYLOAD====================== -->
        <ConnectionTimetableDelivery version="0.1d">
        <ResponseTimestamp>2001-12-17T09:30:47-05:00</ResponseTimestamp>
        <RequestMessageRef>http://www.xmlspy.com</RequestMessageRef>
        <Status>true</Status>
        <ValidUntil>2001-12-17T09:30:47-05:00</ValidUntil>
        <TimetabledFeederArrival>
            <RecordedAtTime>2001-12-17T09:30:47-05:00</RecordedAtTime>
            <ConnectionLinkRef>98789</ConnectionLinkRef>
            <VisitNumber>2</VisitNumber>
            <StopPointName>Erehwon</StopPointName>
            <ConnectingJourney>
                <LineRef>123</LineRef>
                <DirectionRef>OUT</DirectionRef>
                <FramedVehicleJourneyRef>
                    <DataFrameRef>1967-08-13</DataFrameRef>
                    <DatedVehicleJourneyRef>09876</DatedVehicleJourneyRef>
                </FramedVehicleJourneyRef>
                <PublishedLineName>Line 123</PublishedLineName>
                <DirectionName>Outbound</DirectionName>
                <OperatorRef>123</OperatorRef>
                <ProductCategoryRef>School</ProductCategoryRef>
                <ServiceFeatureRef>CyclesPermitted</ServiceFeatureRef>
                <VehicleFeatureRef>LowFloors</VehicleFeatureRef>
                <OriginName>Purgatory</OriginName>
                <DestinationName>Paradise</DestinationName>
                <JourneyNote>from A to B</JourneyNote>
                <OriginAimedDepartureTime>2001-12-17T08:30:47-05:00</OriginAimedDepartureTime>
```

```
                    <DestinationAimedArrivalTime>2001-12-17T10:30:47-
05:00</DestinationAimedArrivalTime>
                    <BlockRef>12345</BlockRef>
                    <CourseOfJourneyRef>89765</CourseOfJourneyRef>
                    <VehicleRef>V987</VehicleRef>
                    <Monitored>true</Monitored>
                    <AimedDepartureTime>2001-12-17T08:35:47-05:00</AimedDepartureTime>
                </ConnectingJourney>
                <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
            </TimetabledFeederArrival>
            <TimetabledFeederArrivalCancellation>
                <RecordedAtTime>2001-12-17T09:30:47-05:00</RecordedAtTime>
                <ConnectionLinkRef>98789</ConnectionLinkRef>
                <VisitNumber>2</VisitNumber>
                    <LineRef>123</LineRef>
                    <DirectionRef>OUT</DirectionRef>
                    <FramedVehicleJourneyRef>
                        <DataFrameRef>1967-08-13</DataFrameRef>
                        <DatedVehicleJourneyRef>09876</DatedVehicleJourneyRef>
                    </FramedVehicleJourneyRef>
                    <PublishedLineName>Line 123</PublishedLineName>
                    <DirectionName>Outbound</DirectionName>
                <Reason>2001-12-17T09:30:47-05:00</Reason>
            </TimetabledFeederArrivalCancellation>
        </ConnectionTimetableDelivery>
    </ServiceDelivery>
```

11 Connection Monitoring Service [CM]

11.1 Purpose

11.1.1 Introduction

The SIRI Connection Monitoring Service exchanges information between different AVMS to coordinate the real-time arrival and departure of PTVs at an interchange through which passengers may make connecting journeys. The departure time of the outgoing ‘distributor’ (or ‘fetcher from’) service may be adjusted to accommodate delays in the incoming ‘feeder to’ service.

The service ensures that the AVMS are in a position to receive all the necessary data concerning the feeder VEHICLEs to allow connection monitoring and dispatch to be carried out. The operational methods of dispatch remain unaffected.

The Service can be used in conjunction with the SIRI Connection Timetable Service to exchange scheduled arrival times for interchanges taking place at the target CONNECTION links.

Transfers between service at a transport interchange may be

a) one-way: the feeder VEHICLE only sets down passengers, and the distributor only picks up, or

b) two-way: that is, both VEHICLEs act as both feeder and distributor to each other, and if either is late for the rendezvous, the other will wait. In the latter case there will be two separate connections, i.e. SERVICE JOURNEY INTERCHANGEs.

The distributor AVMS subscribes to the feeder AVMS to find out about incoming arrivals. Both AVMS may act in both roles, i.e. subscribe to each other in order to coordinate the delayed despatch of distributor services that wait on the feeder service.

The SIRI Connection Monitoring Service comprises the ***ConnectionMonitoringRequest*** message used to specify the contents of request or subscription messages, and two different types of message to deliver the

response; the ***ConnectionMonitoringFeederDelivery*** and the ***ConnectionMonitoringDistributorDelivery*** messages, which return events affecting the feeder or distributor respectively.

The ***ConnectionMonitoringSubscription*** message allows a subscriber to request asynchronous updates for the service: it contains an embedded ***ConnectionMonitoringRequest****,* along with further parameters controlling the asynchronous delivery.

ConnectionMonitoringRequest has topic parameters to filter the information to be returned by time window, VEHICLE JOURNEY, LINE, and DIRECTION, etc, and also policy parameters to control the amount of data returned. In particular it allows the distributor AVMS to specify the type of connection coordination it wishes to use.

The Connection Monitoring service can be used both to monitor journeys whose schedules have been previously exchanged using the timetable service, and to monitor journeys without previous exchange of data; connection coordination may be either ***DatedVehicleJourney***-based or time-based. Both procedures are implemented within the same Connection Monitoring service, using different filters to express which feeder journeys are of interest. The exchange of data is achieved with identical messages and data structures. Differences in the behaviour and results arise from the different request filters used for the two alternative methods.

11.1.2 Dated Vehicle-Journey-Based Connection Protection

DatedVehicleJourney-based Connection Protection requires the prior exchange of the connection times of previously announced VEHICLE JOURNEYs. The prior registration of the VEHICLE JOURNEYs is achieved via a common scheduling system, or by the use of the Connection Timetable Service. The process of agreeing the connections (i.e. SERVICE JOURNEY INTERCHANGEs) can be started directly after importing the current day's schedule into the AVMS system. The number of connections is known in advance.

11.1.3 Time-Based Connection Protection

Time-based Connection Protection is done without the prior exchange of schedule data. Connection agreements are short-term: the service coordinates connections of one or more arbitrary VEHICLEs, arriving as feeders to a CONNECTION link. The VEHICLEs of interest are identified by specifying a feeder LINE, DIRECTION and time window, within which the arrivals of the approaching VEHICLE JOURNEYs shall lie. LINE and DIRECTION shall be known by the requestor, requiring the same extended data management as for the SIRI Connection Timetable service.

11.2 Capability and Permission Matrices

11.2.1 Capability Matrix

The following table presents the set of required and optional capabilities is defined for the Connection Monitoring Service. If the service supports Capability Discovery, the ***ConnectionMonitoringCapabilities-Request*** / ***ConnectionMonitoringCapabilitiesResponse*** message pair can be used to determine the implementation’s capabilities.

Table 88 — ConnectionMonitoringCapabilities Matrix

ConnectionMonitoringCapabilities			*+Structure*	Capabilities describing implementation of Connection Monitoring service.
inherit	***:::***	**1:1**	*See xxx-Capability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***	0:1	*+Structure*	Which optional filtering features are supported?
	DefaultPreviewInterval	0:1	*Positive-DurationType*	Default Preview Interval.
	FilterByConnection-LinkRef	0:1	*xsd:boolean*	Whether results can be filtered by CONNECTION link. Default is *'true'*. SIRI 2.0
	FilterByJourney	**1:1**	*xsd:boolean*	Whether results can be filtered by Journey: required Capability: Fixed as *true*.
	FilterByTime	**1:1**	*xsd:boolean*	Whether results can be filtered by *Time* of connection. Default is *'true'*.
Request Policy	***RequestPolicy***	0:1	*+Structure*	Which optional features of the Request Policy are supported by the service?
	NationalLanguage	**1:***	*xsd:language*	National languages used by service.
	Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
	Coordinates		*choice*	Location coordinate system for results.
	GmlCoordinate-Format	0:1	*SrSName-Type*	Use GML format
	WgsDecimalDegrees	0:1	*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
	ForeignJourneysOnly	0:1	*xsd:boolean*	Whether service returns only foreign journeys. Default is *'false'*.
Sub-scription Policy	***SubscriptionPolicy***	0:1	*+Structure*	Which features of Subscription Policy are supported by service?
	HasIncrementalUpdates	0:1	*xsd:boolean*	Whether incremental updates can be specified for updates Default is *'true'*.
	HasChangeSensitivity	0:1	*xsd:boolean*	Whether change threshold can be specified for updates. Default is *'true'*.
Access Control	***AccessControl***	0:1	*+Structure*	Which optional access control features are supported by service?
	RequestChecking	0:1	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
	CheckOperatorRef	0:1	*xsd:boolean*	If access control is supported, whether access control by OPERATOR is supported. Default is *'true'*.
	CheckLineRef	0:1	*xsd:boolean*	If access control is supported, whether access control by LINE is supported. Default is *'true'*.
	CheckConnection-LinkRef	0:1	*xsd:boolean*	If access control is supported, whether access control by CONNECTION link is supported. Default is *'true'*.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

11.2.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***ConnectionMonitoringCapabilitiesResponse*** response can include the access permissions for the requestor participant to access data (see table below).

Table 89 — ConnectionMonitoring Service — Permissions

ConnectionMonitoringPermission			*+Structure*	Permissions to use implementation of Connection Monitoring service.
In-herit	***:::***	**1:1**	*xxxService-Permissions*	See SIRI Part 2 for Common Permission elements.
Topic	***OperatorPermissions***	0:1	*+Structure*	OPERATOR permissions for participant. See Part 2.
	LinePermissions	0:1	*+Structure*	LINE permissions for participant. See Part 2.
	ConnectionLinkPermissions	0:1	*+Structure*	CONNECTION link permissions for participant.

11.3 The ConnectionMonitoringRequest

11.3.1 ConnectionMonitoringRequest — Element

The ***ConnectionMonitoringRequest*** can be used in both a direct request, and for a subscription (see table below).

The primary ***Topic*** term is the CONNECTION link for which information is to be returned. In addition either a single ***ConnectingTimeFilter*** (for Time-based processing), or one or more ***ConnectingJourneyFilter*** instances (for DATED VEHICLE JOURNEY-based processing) can be specified.

For DATED VEHICLE JOURNEY-based coordination, the ***ConnectingJourneyFilter*** specifies the feeder as well as the aimed arrival time of the VEHICLEs.

For time-based connection coordination the ***ConnectingTimeFilter*** defines the feeder LINEs and/or DIRECTIONs, whose VEHICLE JOURNEYs are to supply schedule deviations and messages. The filter criteria are optional, i.e. it is possible to specify all DIRECTIONs of a LINE, all LINEs in one DIRECTION, or all VEHICLE JOURNEYs approaching a CONNECTION link. The ***ArrivalWindow*** specifies the aimed arrival time window within which the VEHICLE JOURNEYs shall reach the CONNECTION link in order to be reported. The ***StartTime*** element should not be set to a value before the start of the actual data horizon of the server, and ***EndTime*** shall not be after the end of the horizon.

The schedule statuses of the requested VEHICLEs are sent once the ***PreviewTime*** has been reached and after every change (***MonitoredFeederArrival***).

Table 90 — ConnectionMonitoringRequest — Attributes

ConnectionMonitoringRequest			*+Structure*	Request for information about changes to SERVICE JORUNEY INTERCHANGEs at a stop. Made by a distributor AVMS to a feeder AVMS.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Connection Monitoring Service.
Endpoint Properties	***RequestTimestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	MessageIdentifier	0:1	*MessageQualifier*	
Topic	***PreviewInterval***	0:1	*PositiveDuration-Type*	Forward duration for which feeder arrivals should be included, that is, interval before predicted arrival at the stop for which to include arrivals and departure changes: only journeys which will arrive within this time span will be returned.
	ConnectionLinkRef	**1:1**	*→Connection-LinkCode*	Reference to the CONNECTION link for which data is to be returned.
			choice	There must be one time filter or many journey filters.
	a ***Connecting-TimeFilter***	**–1:1**	*+Structure*	Return only journeys for the specified time window.
	b ***Connecting-JourneyFilter***	**–1:***	*+Structure*	Return only the specified journeys.
Request Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: ***NationalLanguage***.
	IncludeTranslations	0:1	*xsd:boolean*	Whether the producer should include any available translations of NLString text elements into multiple languages. If false elements only one value per text element will be provided. +SIRI.2.0 Default is false.
	Connection-MonitoringDetail-Level	0:1	*Connection-Monitoring-DetailLevelEnum*	Default *DetailLevel* if none specified on request. Default is '*normal*'. (+vSIRI 2.0) *minimum* | *basic* | *normal* | *full*
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.3.2 Use of PreviewInterval

The preview interval defines the time interval prior to the aimed arrival of the feeder at the CONNECTION link within which its schedule variance information should be transmitted. If the VEHICLE is not en-route at this point, the timetabled information is transmitted.

The preview interval reduces amount of data within the time frame in which connection monitoring is active for the distributor. This is particularly significant in the case of longer journeys (e.g. long-distance travel).

For Time-Based connection protection, the preview interval can be used to create long-term subscriptions for CONNECTION links that involve only a few feeder VEHICLEs and a large number of distributor VEHICLEs. In such cases, without a preview time, Journey-Based connection protection would yield a large number of subscriptions (one per distributor VEHICLE JOURNEY). For such CONNECTION links it is better to use a time-based subscription for the entire day, from which all distributor information is then derived.

The preview interval is optional. If not specified, the time elements of the subscription are valid (VEHICLE JOURNEY-based), or the ***EarliestArrivalTime***.

11.3.3 ConnectingTimeFilter — Element

The ***ConnectingTimeFilter*** is used for Time-based connection protection to indicate the time window to be monitored (see table below). A LINE and DIRECTION may also be specified.

Table 91 — ConnectingTimeFilter — Attributes

Filter	***ConnectingTimeFilter***		*+Structure*	Return only journeys for the specified time window.
	LineRef	**1:1**	*→LineCode*	Filter for the feeder LINE, which is to supply data.
	DirectionRef	**1:1**	*→DirectionCode*	Filter for the feeder DIRECTION for which data is to be supplied.
	EarliestArrivalTime	0:1	*xsd:dateTime*	Earliest managed arrival time of VEHICLE at the CONNECTION link.
	LatestArrivalTime	0:1	*xsd:dateTime*	Latest managed arrival time of VEHICLE at the CONNECTION links.

11.3.4 ConnectingJourneyFilter — Element

The ***ConnectingJourneyFilter*** is used for the DATED VEHICLE JOURNEY-based connection protection to indicate the journey or journeys to be monitored (see table below).

Table 92 — ConnectingJourneyFilter — Attributes

Filter	***ConnectingJourneyFilter***		*+Structure*	Return only the specified journeys.
	DatedVehicle-JourneyRef	**1:1**	*→DatedVehicle-JourneyCode*	A reference to a DATED VEHICLE JOURNEY.
	VisitNumber	0:1	*VisitNumber-Type*	Sequence of visit to stop within VEHICLE JOURNEY. Increases monotonically but not necessarily sequentially.
	Timetabled-ArrivalTime	**1:1**	*xsd:dateTime*	Timetabled arrival time of VEHICLE at the CONNECTION link.

11.3.5 ConnectionMonitoringRequest — Example

The following is an example of a ***ConnectionMonitoringRequest***.

```
<ServiceRequest>
    <!--======ENDPOINT REFERENCES================================-->
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <RequestorRef>NADER</RequestorRef>
    <!--========JOURNEY FILTER ==================================== -->
    <ConnectionMonitoringRequest version="0.1d">
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--========TOPIC ==================================== -->
        <ConnectionLinkRef>EH00001</ConnectionLinkRef>
        <ConnectingJourneyFilter>
            <DatedVehicleJourneyRef>ABC56789</DatedVehicleJourneyRef>
            <TimetabledArrivalTime>2004-12-17T10:00:47-05:00</TimetabledArrivalTime>
        </ConnectingJourneyFilter>
    </ConnectionMonitoringRequest>
    <!--========TIME FILTER ==================================== -->
    <ConnectionMonitoringRequest version="0.1d">
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--========TOPIC ==================================== -->
        <PreviewInterval>PT10M</PreviewInterval>
```

```
        <ConnectionLinkRef>EH00002</ConnectionLinkRef>
        <ConnectingTimeFilter>
            <LineRef>LINE77</LineRef>
            <DirectionRef>OUT</DirectionRef>
        </ConnectingTimeFilter>
    </ConnectionMonitoringRequest>
</ServiceRequest>
```

11.4 The ConnectionMonitoringSubscriptionRequest

11.4.1 ConnectionMonitoringSubscriptionRequest — Element

The ***ConnectionMonitoringSubscriptionRequest*** requests the asynchronous delivery of the information described by a ***ConnectionMonitoringRequest*** (see Table 93 below).

The ***InitialTerminationTime*** shall be equal to or later than ***LatestArrivalTime*** on the request.

Table 93 — ConnectionMonitoringSubscriptionRequest — Attributes

ConnectionMonitoringSubscription-Request			*+Structure*	Request for a subscription to the Connection Monitoring Service.
Identity	***SubscriberRef***	0:1	*→ParticipantCode*	See SIRI Part 2 Common ***SubscriptionRequest*** parameters.
	SubscriptionIdentifier	**1:1**	*SubscriptionQualifier*	
Lease	***InitialTerminationTime***	**1:1**	*xsd:dateTime*	
Request	***ConnectionMonitoring-Request***	**1:1**	*+Structure*	See ***ConnectionMonitoringRequest***.
Policy	***ChangeBeforeUpdates***	0:1	*PositiveDurationType*	The amount of change to the arrival time that can happen before an update is sent (i.e. if ***ChangeBeforeUpdate*** is set to 2 min, the subscriber will not be told that a VEHICLE is 30 s delayed – an update will only be sent when the VEHICLE is at least 2 min delayed). Default is zero – all changes will be sent regardless according to the processing cycle of the Producer. Optional SIRI capability: ***ChangeBeforeUpdate*s**.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.4.2 Sensitivity Threshold

When a subscription is first setup, the current connection data is transmitted, thereafter only changes are sent, as indicated by to the specified ***ChangeBeforeUpdates***. Change before update is an option capability.

Data is deemed to have changed if at least one of the following pieces of information has changed:

— *Variation from schedule adherence:* the planned SERVICE INTERCHANGE cannot take place because the feeder does not reach the CONNECTION link, or reaches it too late;

— *Arrival at CONNECTION link*: a change remains active until the data assigned to the subscription is explicitly retrieved, which resets the status;

— *Change of platform*: a platform change is a discrete event that is communicated as soon as it is known.

In order to avoid constant updates containing insignificant changes to the predicted arrival time, a sensitivity threshold value can be assigned within the subscription. This defines the time span, after which the system

regards the variation in schedule adherence significant enough to demand communication. With a sensitivity of 2 min, for example, changes in deviations of 2, 4, 6 min, etc. are reported. The change is with respect to the last transmitted arrival time. However, use of a sensitivity value is only a recommendation and can also be ignored by the data producing system.

11.4.3 ConnectionMonitoringSubscriptionRequest — Example

The following is an example of a ***ConnectionMonitoringSubscriptionRequest***.

```
    <SubscriptionRequest>
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <RequestorRef>NADER</RequestorRef>
        <ConnectionMonitoringSubscriptionRequest>
            <SubscriberRef>NADER</SubscriberRef>
            <SubscriptionIdentifier>000234</SubscriptionIdentifier>
            <InitialTerminationTime>2004-12-17T09:30:47-05:00</InitialTerminationTime>
            <ConnectionMonitoringRequest version="0.1d">
                <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
                <!--========TOPIC ==================================== -->
                <ConnectionLinkRef>EH00001</ConnectionLinkRef>
                <ConnectingJourneyFilter>
                    <DatedVehicleJourneyRef>ABC56789</DatedVehicleJourneyRef>
                    <TimetabledArrivalTime>2004-12-17T10:00:47-
05:00</TimetabledArrivalTime>
                </ConnectingJourneyFilter>
            </ConnectionMonitoringRequest>
        </ConnectionMonitoringSubscriptionRequest>
    </SubscriptionRequest>
```

11.5 ServiceDelivery with Connection Monitoring Deliveries — Element

11.5.1 Introduction

One or more ***ConnectionMonitoringFeederDelivery*** or ***ConnectionMonitoringDistributorDelivery*** elements may be returned as part of a SIRI ***ServiceDelivery***, with a common ***ResponseTimestamp*** (see table below).

Table 94 — ServiceDelivery / ConnectionMonitoringDelivery — Attributes

ServiceDelivery			*+Structure*	See SIRI Part 2 ***ServiceDelivery***
HEADER	*:::*	**1:1**	See ServiceDelivery	
Payload	***ConnectionMonitoring-FeederDelivery***	**1:***	*+Structure*	See ***ConnectionMonitoringFeederDelivery***.
	ConnectionMonitoring-DistributorDelivery		*+Structure*	See ***ConnectionMonitoringDistributor-Delivery***.

11.5.2 The ConnectionMonitoringFeederDelivery — Element

11.5.2.1 General

A ***ConnectionMonitoringFeederDelivery*** is made up of a variable number of ***MonitoredFeederArrival*** elements), and/or ***MonitoredFeederArrivalCancellation*** elements, representing new connections, changes, or loss of planned interchanges respectively (see Table 95 below).

Table 95 — ConnectionMonitoringFeederDelivery — Attributes

ConnectionMonitoringFeederDelivery			*+Structure*	Delivery of feeder arrivals (i.e. incoming SERVICE JOURNEY INTERCHANGEs) for ***Connection-Monitoring*** Service.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of CM Service. Fixed e.g., '1.1a'.
LEADER	***:::***	**1:1**	xxx***Service-Delivery***	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***MonitoredFeeder-Arrival***	0:*	*+Structure*	A change to a feeder arrival for the CONNECTION link. See ***MonitoredFeederArrival*** element.
	MonitoredFeeder-ArrivalCancellation	0:*	*+Structure*	A cancellation of a managed feeder SERVICE JOURNEY INTERCHANGE to CONNECTION link. See ***MonitoredFeederArrival*** element.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

11.5.2.2 MonitoredFeederArrival — Element

11.5.2.2.1 Overview

Each ***MonitoredFeederArrival*** describes a change to an incoming VEHICLE JOURNEY to a connection (see Table 96 below).

Table 96 — MonitoredFeederArrival — Attributes

MonitoredFeederArrival			*+Structure*	A change to the managed feeder connection, i.e. incoming SERVICE JOURNEY INTERCHANGE at a given CONNECTION link.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which ***FeederArrival*** was recorded.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item within data horizon of producer. Can be used to reference Item later, e.g. for server side cleardown.
Currency	***ValidUntilTime***	0:1	*xsd:dateTime*	Time until which data is valid. +SIRI v2.0
Feeder Inter-change Identity	***InterchangeRef***	0:1	*→Interchange-Code*	Reference to a SERVICE JOURNEY INTERCHANGE between two journeys for which data is being returned.
	ConnectionLinkRef	**1:1**	*→Connection-LinkCode*	Reference to a CONNECTION link or ACCESS ZONE for which data is to be returned.
	StopPointRef	0:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT. Usually this will be the given by the CONNECTION link for which data is to be returned.
	VisitNumber	0:1	*VisitNumber-Type*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to the same stop, the ***VisitNumber*** is used to distinguish each separate visit.
	Order	0:1	*xsd:positive-Integer*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPointName	0:*	*NLString*	Feeder stop name of CONNECTION link. One per language (Unbounded 0:* since +SIRI v2.0).

	ClearDownRef	0:1	*→CleardownCode*	Cleardown identifier of SERVICE JOURNEY INTERCHANGE arrival that is being deleted.
Journey Info	***FeederJourney***	**1:1**	*InterchangeJourneyStructure*	Provides information about the MONITORED VEHICLE JOURNEY which this VEHICLE is running. See Feeder element earlier.
Real-time call	***VehicleAtStop***	0:1	*xsd:boolean*	Whether VEHICLE is at stop. If absent unknown.
	NumberOfTransfer-Passengers	0:1	*xsd:nonNegativeInteger*	Number of passengers who wish to transfer at the CONNECTION link. If absent, not known.
	AimedArrivalTime	0:1	*xsd:dateTime*	Aimed Arrival Time
Call time	***ExpectedArrival-Time***	**1:1**	*xsd:dateTime*	Estimated arrival time of VEHICLE at the CONNECTION link.
	ArrivalPlatform-Name	**0:1**	*NLString*	Bay or platform name.
	SuggestedWait-DecisionTime	0:1	*xsd:dateTime*	Latest time by which the feeder needs information about the connection from the distributor as to whether it will wait and for how long. +SIRI v2.0
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.5.2.2.2 MonitoredFeederArrival / FeederJourney — Element

The ***FeederJourney*** element is an instance of the ***InterchangeJourney*** structure that can be populated with information about the Feeder Journey (see Table 97 below).

Table 97 — FeederJourney — Attributes

FeederJourney			*InterchangeJourneyStructure*	Information about a feeder VEHICLE JOURNEY that arrives at a CONNECTION link
VehicleJourney-Identity	***LineRef***	**1:1**	*→LineCode*	Identifier for the LINE of the feeder journey.
	DirectionRef	**1:1**	*→DirectionCode*	Identifier of the relative DIRECTION the VEHICLE is running along the LINE, for example, "in" or "out", “clockwise”. Distinct from a destination.
	FramedVehicle-JourneyRef	0:1	*+Structure*	A compound reference to the DATED VEHICLE JOURNEY that the VEHICLE is making, unique within data horizon of producer. See SIRI Part 2.
JourneyPatternInfo	***:::***	0:1	*JourneyPatternInfoGroup*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
VehicleJourneyInfo	***:::***	0:1	*VehicleJourneyInfoGroup*	See SIRI Part 2 ***VehicleJourneyInfoGroup***.
Operational Info	***:::***	0:1	*Operational_InfoGroup*	See SIRI Part 2 ***OperationalInfoGroup***.
DisruptionGroup	***:::***	0:1	*DisruptionGroup*	See SIRI Part 2 ***VehicleDisruptiomInfoGroup*** Disruption data associated with feeder.
Progress	***Monitored***	0:1	*xsd:boolean*	Whether there is real-time information available for journey, if not present, not known.
Call Times	***AimedDeparture-Time***	0:1	*xsd:dateTime*	Aimed arrival time at the CONNECTION link. NB this usage is confusing and arises because a commed structure is shared for arrival and departure interchange journeys
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.5.2.3 MonitoredFeederArrivalCancellation — Element

Each ***MonitoredFeederArrivalCancellation*** (see Table 98 below) describes the loss of an inbound connection.

Table 98 — MonitoredFeederArrivalCancellation — Attributes

MonitoredFeederArrivalCancellation			*+Structure*	Cancellation of previous connection
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which ***FeederArrivalCancellation*** was recorded.
Identity	***ItemRef***	0:1	*ItemIdentifier*	Reference to a previous ***FeederArrival*** which this item cancels.
Feeder Inter-change-Identity	***InterchangeRef***	0:1	*→InterchangeCode*	Reference to a SERVICE JOURNEY INTERCHANGE between two journeys for which data is being returned.
	ConnectionLinkRef	**1:1**	*→ConnectionLink-Code*	Reference to the CONNECTION link for which data is to be returned.
	StopPointRef	0:1	*→StopPointCode*	Reference to a SCHEDULED STOP POINT. Usually this will be given by the CONNECTION link for which data is to be returned.
	VisitNumber	0:1	*VisitNumberType*	For JOURNEY PATTERNs that involve repeated visits by a VEHICLE to the same stop, the ***VisitNumber*** is used to distinguish each separate visit.
	Order	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
	StopPoint~Name	0:*	*NLString*	Name of feeder SCHEDULED STOP POINT of CONNECTION link. One per language (Unbounded 0:* since +SIRI v2.0).
Journey Info	***LineRef***	**1:1**	*→LineCode*	Reference to a LINE.
	DirectionRef	**1:1**	*→DestinationCode*	Reference to a DIRECTION.
	VehicleJourneyRef	**1:1**	*+FramedVehicle-JourneyRefStructure*	Reference to the feeder VEHICLE JOURNEYs. See SIRI Part 2.
Journey-Pattern-Info	***:::***	0:1	*JourneyPatternInfo-Group*	See SIRI Part 2 ***JourneyPatternInfoGroup***.
Info	***Reason***	0:*	*NLString*	Reason for failure of SERVICE JOURNEY INTERCHANGE. One per language (Unbounded 0:* since +SIRI v2.0).
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

Loss of a feeder indicates the cancellation of the visit to the stop of the CONNECTION link, or a severely delayed arrival at the connection stop. This can lead to the distributor VEHICLE abandoning the connection. There are several possible reasons for the loss of a feeder, e.g. loss/cancellation of the feeder VEHICLE, diversion of the feeder LINE/DIRECTION, disruption of a section of LINE, or short working.

The text elements (LINE, DIRECTION) are for informative purposes only for the dispatcher, as the VEHICLE JOURNEY is uniquely referenced via the ***FeederVehicleJourneyRef***.

The VEHICLE JOURNEY itself shall not be deleted from the subscription on the basis of the failure message from the feeder. Following a distributor reset after a loss of data (***DataSupplyAll***), the failure message shall be sent again in place of a ***MonitoredFeederArrival***.

11.5.2.4 ConnectionMonitoringFeederDelivery — Example

The following is an example of a ***ConnectingMonitoringFeederDelivery***.

```
<ServiceDelivery>
    <!--=======HEADER================================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <!--=======PAYLOAD================================================ -->
    <ConnectionMonitoringFeederDelivery version="0.1d">
        <ResponseTimestamp>2001-12-17T09:30:47.0Z</ResponseTimestamp>
        <RequestMessageRef>12345</RequestMessageRef>
        <Status>true</Status>
    <!--=======Functional SERVICE PAYLOAD========================== -->
        <MonitoredFeederArrival>
            <RecordedAtTime>2001-12-17T09:30:47-05:00</ RecordedAtTime>
            <ConnectionLinkRef>HLKT00023</ConnectionLinkRef>
            <VisitNumber>001</ VisitNumber>
            <ConnectingJourney>
                <LineRef>ABC</LineRef>
                <DirectionRef>Out</ DirectionRef>
            </ConnectingJourney>
            <AimedArrivalTime>2001-12-17T09:30:47-05:00</AimedArrivalTime>
        </ MonitoredFeederArrival>
    </ConnectionTimetableFeederDelivery>
</ServiceDelivery>
```

11.5.3 The ConnectionMonitoringDistributorDelivery — Element

11.5.3.1 Introduction

11.5.3.1.1 General

In addition to the events that can be transmitted from the feeder to the distributor, there are further events that can be transmitted in the opposite direction, i.e. from the distributor to the feeder. These may be used to provide information to passengers of the feeder VEHICLE JOURNEYs. All distributor-to-feeder events are optional. Whether or not a system sends these messages depends on the operating agreement.

The distributor events are marked with the identifiers of one or more feeder VEHICLE JOURNEYs, enabling a notification of the appropriate feeders to be made.

A separate subscription is not required to obtain these Distributor events (and indeed it is not possible to get them without the Feeder Events).

The following distributor-to-feeder messages are provided:

— Change to the dispatch status of the distributor (prolonged waiting time);

— Change of departure location of the distributor at the CONNECTION link;

— Cancellation of the distributor.

A ***ConnectionMonitoringDistributorDelivery*** (see table below) is made up of a variable number of ***Wait-ProlongedDeparture*** elements, ***StoppingPositionChangedDeparture*** elements, and/or ***Distributor-DepartureCancellation*** elements, representing delays, platform changes or loss of planned interchanges respectively.

Table 99 — ConnectionMonitoringDistributorDelivery — Attributes

ConnectionMonitoringDistributor-Delivery			*+Structure*	Timetabled Delivery of Feeder connections for Connection Timetable Service.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of Connection Timetable Service. Fixed e.g., '1.1a'.
LEADER	***:::***	**1:1**	xxx***ServiceDelivery***	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***WaitProlonged-Departure***	0:*	*+Structure*	A managed feeder connection to the CONNECTION link; see ***WaitProlonged-Departure***.
	StoppingPosition-ChangedDeparture	0:*	*+Structure*	A change to the stopping position of an inbound journey.
	DistributorDeparture-Cancellation	0:*	*+Structure*	A cancellation of a managed feeder at CONNECTION link.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.5.3.1.2 DistributorInfoGroup

The ***DistributorInfoGroup*** provides data to identify operational entities associated with a distributor journey leaving from an interchange (see Table 100 below). It is common to all three delivery payload elements.

Table 100 — DistributorInfoGroup

Distributor Inter-change_ Identity	***InterchangeRef***	0:1	*→InterchangeCode*	Reference to the SERVICE JOURNEY INTERCHANGE between two journeys for which data is being returned.
	ConnectionLinkRef	**1:1**	*→Connection-LinkCode*	Reference to the CONNECTION link for which data is to be returned.
	StopPointRef	0:1	*→StopPoint-Code*	Reference to a SCHEDULED STOP POINT within CONNECTION from which distributor leaves.
	Distributor-VisitNumber	0:1	*VisitNumber-Type*	Order of visit to a stop within JOURNEY PATTERN of distributor VEHICLE JOURNEY.
	DistributorOrder	0:1	*xsd:positiveInteger*	For implementations for which the overall Order within JOURNEY PATTERN is not used for ***VisitNumber***, (i.e. if ***VisitNumberIsOrder*** is *false*) then can be used to associate the overall Order as well if useful.
Journey Info	***DistributorJourney***	**1:1**	*Connecting-JourneyStructure*	Information about the connecting distributor VEHICLE JOURNEY. See ***DistributorJourney*** element.
Feeder Info	***FeederVehicle-JourneyRef***	0:*	*FramedVehicle-JourneyRef-Structure*	A reference to one or more feeder VEHICLE JOURNEYs. See SIRI Part 2.

11.5.3.2 WaitProlongedDeparture — Element

11.5.3.2.1 Overview

An important function of Connection Protection is the ability to hold back a distributor VEHICLE (i.e. prolonged waiting) to allow passengers to transfer from delayed feeders. To achieve this a ***WaitProlongedDeparture*** (see Table 101 below) shall be communicated back to the feeder VEHICLEs to inform the passengers about the new departure time of the distributor (e.g. via on-board displays in the VEHICLE). A prolonged wait is

reported to the feeder with the ***WaitProlongedDeparture*** message within a ***ConnectionMonitoring-DistributorDelivery***.

Table 101 — WaitProlongedDeparture — Attributes

WaitProlongedDeparture			*+Structure*	An action to delay the distributor until a specified time.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which Item was recorded.
Distrib-utorInfo	***:::***	**1:1**	*DistributorInfo-Group*	See above ***DistributorInfoGroup.***
Change	***Expected-DepartureTime***	0:1	*xsd:dateTime*	Estimated departure time of VEHICLE from the SERVICE JOURNEY INTERCHANGE.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.5.3.2.2 Distributor Journey — Element

The ***DistributorJourney*** element is an instance of the ***ConnectingJourney*** structure that can be populated with information about the Distributor Journey. See ***FeederJourney*** for details.

11.5.3.3 StoppingPositionChangedDeparture — Element

The ***StoppingPositionChangedDeparture*** function is used to provide the passenger with information about the new platform position of the distributor VEHICLE if the distributor changes its planned stopping position within the CONNECTION link.

Each ***StoppingPositionChangedDeparture****,* describes a change to the stopping position of a distributor (see Table 102 below).

Table 102 — StoppingPositionChangedDeparture — Attributes

StoppingPositionChangedDeparture			*+Structure*	A change to the stopping position of an inbound SERVICE JOURNEY INTERCHANGE.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which Item was recorded.
Distrib-utorInfo	***:::***	**1:1**	*DistributorInfoGroup*	See above *DistributorInfoGroup.*
Change	***ChangeNote***	0:*	*NLString*	Description of the revised stopping position of the distributor in the CONNECTION zone. One per language (Unbounded 1:* since +SIRI v2.0).
	NewLocation	0:1	*→Location*	New location for distributor VEHICLE.
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.5.3.4 DistributorDepartureCancellation — Element

The cancellation of a distributor means the cancellation of the corresponding connection, i.e. incoming SERVICE JOURNEY INTERCHANGE. The ***DistributorDepartureCancellation*** (see table below) message is used to signal the cancellation of the onward connection to the passengers in the feeder VEHICLEs.

The cancellation affects the entire subscription, i.e. all previously reported feeders for the distributor's journey.

To stop the continuing transmission of feeder ***MonitoredFeederArrival*** instances, the associated subscription shall be deleted by the distributor.

Table 103 — DistributorDepartureCancellation — Attributes

DistributorDepartureCancellation			*+Structure*	Cancellation of previous monitored connection.
Log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which Item was recorded.
Distributor-Info	***:::***	**1:1**	*DistributorInfoGroup*	See above *DistributorInfoGroup*.
Call time	***Reason***	0:*	*NLString*	Reason for failure of SERVICE JOURNEY INTERCHANGE. One per language (Unbounded 1:* since +SIRI v2.0).
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

11.5.3.5 ConnectionMonitoringDistributorDelivery — Example

The following is an example of a ***ConnectionMonitoringDistributorDelivery***.

```
<ServiceDelivery>
    <!--=======HEADER==================================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <!--=======PAYLOAD================================================= -->
    <ConnectionMonitoringDistributorDelivery version="0.1d">
        <ResponseTimestamp>2001-12-17T09:30:47.0Z</ResponseTimestamp>
        <RequestMessageRef>12345</RequestMessageRef>
        <Status>true</Status>
        <!--========WAIT PROLONGED DEPARTURE ============================ -->
        <WaitProlongedDeparture>
            <RecordedAtTime>2001-12-17T09:30:47-05:00</RecordedAtTime>
            <ConnectionLinkRef>HLKT00023</ConnectionLinkRef>
            <DistributorVisitNumber>001</DistributorVisitNumber>
            <DistributorJourney>
                <LineRef>ABC</LineRef>
                <DirectionRef>Out</DirectionRef>
            </DistributorJourney>
            <FeederVehicleJourneyRef>
                <DataFrameRef>2001-12-17</ DataFrameRef>
                <DatedVehicleJourneyRef>8776654986</DatedVehicleJourneyRef>
            </FeederVehicleJourneyRef>
            <ExpectedDepartureTime>2001-12-17T09:30:47-05:00</ExpectedDepartureTime>
        </WaitProlongedDeparture>
        <!--========STOPPING POSITION CHANGED DEPARTURE =============== -->
        <StoppingPositionChangedDeparture>
            <RecordedAtTime>2001-12-17T09:30:47-05:00</RecordedAtTime>
            <ConnectionLinkRef>HLKT00022</ConnectionLinkRef>
            <DistributorVisitNumber>001</DistributorVisitNumber>
            <DistributorJourney>
                <LineRef>A11C</LineRef>
                <DirectionRef>OUT</DirectionRef>
                <FramedVehicleJourneyRef>
                    <DataFrameRef>2001-12-17</DataFrameRef>
                    <DatedVehicleJourneyRef>09876</DatedVehicleJourneyRef>
                </FramedVehicleJourneyRef>
                <PublishedLineName>Line A11C</PublishedLineName>
                <DirectionName>Outbound</DirectionName>
                <OperatorRef>1s23</OperatorRef>
                <ProductCategoryRef>School</ProductCategoryRef>
                <ServiceFeatureRef>CyclesPermitted</ServiceFeatureRef>
                <VehicleFeatureRef>LowFloors</VehicleFeatureRef>
                <OriginName>Purgatory</OriginName>
                <DestinationName>Paradise</DestinationName>
                <JourneyNote>from A to B</JourneyNote>
```

```
                    <OriginAimedDepartureTime>2001-12-17T08:30:47-05:00</OriginAimedDepartureTime>
                    <DestinationAimedArrivalTime>2001-12-17T10:30:47-
05:00</DestinationAimedArrivalTime>
                    <BlockRef>12345</BlockRef>
                    <CourseOfJourneyRef>89765</CourseOfJourneyRef>
                    <VehicleRef>V987</VehicleRef>
                    <Monitored>true</Monitored>
                    <AimedDepartureTime>2001-12-17T09:34:47-05:00</AimedDepartureTime>
                </DistributorJourney>
                <FeederVehicleJourneyRef>
                    <DataFrameRef>2001-12-17</DataFrameRef>
                    <DatedVehicleJourneyRef>87766545677</DatedVehicleJourneyRef>
                </FeederVehicleJourneyRef>
                <ChangeNote>Will now leave from platform 6 </ChangeNote>
            </StoppingPositionChangedDeparture>

            <!--=======DEPARTURE CANCELLATION EVENT ================ -->
            <DistributorDepartureCancellation>
                <RecordedAtTime>2001-12-17T09:30:47-05:00</RecordedAtTime>
                <ConnectionLinkRef>987259</ConnectionLinkRef>
                <DistributorVisitNumber>2</DistributorVisitNumber>
                <DIstributorJourney>
                    <LineRef>123</LineRef>
                    <DirectionRef>OUT</DirectionRef>
                    <PublishedLineName>Line 123</PublishedLineName>
                    <DirectionName>Outbound</DirectionName>
                </DistributorJourney>
                <FeederVehicleJourneyRef>
                    <DataFrameRef>2001-12-17</DataFrameRef>
                    <DatedVehicleJourneyRef>09867</DatedVehicleJourneyRef>
                </FeederVehicleJourneyRef>
                <Reason>Short staff</Reason>
            </DistributorDepartureCancellation>
        </ConnectionMonitoringDistributorDelivery>
    </ServiceDelivery>
```

12 General Message Service [GM]

12.1 Purpose

The SIRI General Message service is used to transmit messages between the participants. The data to be published will typically be informative messages such as travel news and other operational advice, entered or forwarded into the system, normally by a control centre. The General Message service can segregate different types of informative message into separate information channels; each info channel can be assigned to a different operational message group type (errors, messages, warnings, traffic information, operational messages, etc.). A subscription may be to a specific info channel, allowing data for an individual channel to be polled separately.

The General Message Service comprises the ***GeneralMessageRequest*** message used to specify the contents of request or subscription messages, and the ***GeneralMessageDelivery*** message used to deliver the response. The ***GeneralMessageSubscription*** message allows a subscriber to request asynchronous updates for the service: it contains an embedded ***GeneralMessageRequest***.

GeneralMessageRequest has topic parameters to filter the information by channel.

The ***GeneralMessageDelivery*** returns information in an extensible format as one or more ***InfoMessage*** instances.

General Messages can be revoked if the message loses its validity before the planned time. For example, if a message has been sent to set up a special purpose VEHICLE with a planned duration, the message can be revoked (using an ***InfoMessageCancellation***) if the VEHICLE finishes early. It thus allows SIRI to be used as a conduit for the conventional store and forward transmission of news and disruption messages.

12.2 Reference Data

12.2.1 Use of Reference Data

In order to be able to assign messages to different operational message categories, it is necessary for participants to agree ***InfoChannelIdentifier*** values, which are then referenced in messages as ***InfoChannel-Refs***. The identifiers define a mutual understanding of the message categorization, for example 'warning', 'sever disruption'. The ***InfoChannelRefs*** are managed within the data management of the participant systems. If structured data is to be exchanged between the control centres, a ***FormatRef*** shall also be agreed on by the various participants.

An optional discovery service is described which can be used to obtain the available channels.

12.2.2 Message Formats

The message service can transmit informative messages containing three types of data:

— Simple free format text.

— Text with structured characteristics according to an arbitrary format.

— Fully structure content as defined by an arbitrary XML subschema.

XML content has a structure specified by a schema; other formats can be applied to plain text to permit the subsequent interpretation of it by another computer, for example CSV (comma separated value) format, a tabular format in which the column values are separated by commas and the rows by line-feed characters.

In order to be able to use and detect different formats at run time, a ***FormatRef*** is used to declare the mark-up of each message. For XML content, the URL of the content schema should be used as the ***FormatRef***. If no ***FormatRef*** is specified, then the message is interpreted as plain text without formatting. The format is metadata, and is therefore passed as an attribute.

12.3 Capability and Permission Matrices

12.3.1 Capability Matrix

The following table provides a set of required and optional capabilities is defined for the General Message service. If the service supports Capability Discovery, the ***GeneralMessageCapabilitiesRequest*** / ***General-MessageCapabilitiesResponse*** message pair can be used to determine the implementation's capabilities.

Table 104 — GeneralMessageCapabilities Matrix

GeneralMessageCapabilities				*+Structure*	Capabilities describing implementation of General Message service.
inherit	***:::***		**1:1**	*See xxxCapability-Response*	See SIRI Part 2 for Common Capability attributes.
Topic	***TopicFiltering***		0:1	*+Structure*	Which optional filtering features are supported?
		DefaultPreviewI nterval	**1:1**	*Positive-DurationType*	Default preview Interval. Default is 60 min
		FilterByInfo-Channel	0:1	*xsd:boolean*	Whether results can be filtered by Info Channel. Required Capability: Default is *'true'*.

Request Policy	***RequestPolicy***	0:1	*+Structure*	Which optional features of the Request Policy are supported by the service?
	National-Language	**1:***	*xsd:language*	National languages used by service.
	Translations	0:1	*xsd:boolean*	Whether the producer supports translations. SIRI 2.0 Default is false.
	Coordinates		*choice*	Location coordinate system for results.
	GmlCoordinateFormat	0:1	*SrSNameType*	Use GML format
	WgsDecimalDegrees	0:1	*EmptyType*	Default coordinate data system is WGS 84 latitude and longitude.
Access Control	***AccessControl***	0:1	*+Structure*	Which optional Access Control features are supported by service?
	Request-Checking	0:1	*xsd:boolean*	Whether access control of requests is supported. Default is *'false'*.
	CheckInfo-Channel	0:1	*xsd:boolean*	If access control is supported, whether access control by *InfoChannel* is supported. Default is *'true'*.
any	***Extensions***	0:1	xsd:any*	Placeholder for user extensions.

12.3.2 Permission Matrix

If the implementation supports both Capability Discovery and Access Controls, then the ***GeneralMessage-CapabilitiesResponse*** message can include the access permissions for the requestor participant to access data (see Table 105 below).

Table 105 — GeneralMessagePermission Matrix

GeneralMessagePermission			*+Structure*	Permissions to use implementation of General Message service.
In-herit	***:::***	**1:1**	*xxxServicePer-missions*	See SIRI Part 2 for Common Permission elements.
Topic	***InfoChannelPermissions***	0:1	*+Structure*	Infochannel permissions for participant. See Part 2.

12.4 The GeneralMessageRequest

12.4.1 GeneralMessageRequest — Element

The ***GeneralMessageRequest*** (see table below) can be used in both a direct request, and for a subscription.

The primary *Topic* term is the Info Channel or channels for which information is to be returned.

Table 106 — GeneralMessageRequest — Permissions

GeneralMessageRequest			*+Structure*	Request for information about messages.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of General Message Service.
Endpoint Properties	***Request-Timestamp***	**1:1**	*xsd:dateTime*	See SIRI Part 2 Common properties of SIRI Functional Service Requests.
	Message-Identifier	0:1	*MessageQualifier*	
Topic	***InfoChannelRef***	0:*	*→InfoChannelCode*	Reference to the Channel for which data is to be returned. If absent, all channels.
Request Policy	***Language***	0:1	*xml:lang*	Preferred language in which to return text values. Optional SIRI capability: *NationalLanguage.*
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

12.4.2 GeneralMessageRequest — Example

The following is an example of a ***GeneralMessageRequest*** to receive just 'WARNING' messages.

```
<ServiceRequest">
    <!--======ENDPOINT REFERENCES================================-->
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
    <GeneralMessageRequest version="1.0”>
        <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!--=======TOPIC ================================== -->
        <InfoChannelRef>WARNING</InfoChannelRef>
    </GeneralMessageRequest>
</ServiceRequest>
```

12.5 The GeneralMessageSubscriptionRequest

12.5.1 GeneralMessageSubscriptionRequest — Element

The ***GeneralMessageSubscriptionRequest*** (see Table 107 below) requests the asynchronous delivery of the information described by a ***GeneralMessageRequest***.

Table 107 — GeneralMessageSubscriptionRequest — Attributes

GeneralMessageSubscriptionRequest			*+Structure*	Request for a subscription to the SIRI ***GeneralMessage*** Service.
Identity	***SubscriberRef***	0:1	*→ParticipantCode*	See SIRI Part 2 Common ***SubscriptionRequest*** parameters.
	SubscriptionIdentifier	**1:1**	*SubscriptionQualifier*	
Lease	***InitialTerminationTime***	**1:1**	*xsd:dateTime*	
Request	***GeneralMessageRequest***	**1:1**	*+Structure*	See ***GeneralMessageRequest***.
	Extensions	0:1	xsd:any*	Placeholder for user extensions.

12.5.2 GeneralMessageSubscriptionRequest — Example

The following is an example of a ***GeneralMessageSubscriptionRequest***. This example subscribes to receive all info channels.

```
<SubscriptionRequest>
    <!--======ENDPOINT REFERENCES================================-->
    <RequestorRef>NADER</RequestorRef>
    <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
        <!-- Subscription 1 for VIS123 to Place 975 -->
    <GeneralMessageSubscriptionRequest>
        <SubscriptionIdentifier>00000456</SubscriptionIdentifier>
        <InitialTerminationTime>2004-12-17T09:30:47-05:00</InitialTerminationTime>
        <!-- ======ENDPOINT REFERENCE================================-->
        <GeneralMessageRequest version="0.1d”>
            <RequestTimestamp>2004-12-17T09:30:47-05:00</RequestTimestamp>
            <!--=======TOPIC ================================== -->
        </GeneralMessageRequest>
    </GeneralMessageSubscriptionRequest>
</SubscriptionRequest>
```

12.6 The GeneralMessageDelivery

12.6.1 Introduction

The ***GeneralMessageDelivery*** returns the current ***InfoMessage*** instances for a channel.

12.6.2 ServiceDelivery with a GeneralMessageDelivery

One or more ***GeneralMessageDelivery*** (see Table 108 below) elements may be returned as part of a SIRI ***ServiceDelivery***, with a common ***ResponseTimestamp***.

Table 108 — ServiceDelivery / GeneralMessageDelivery — Attributes

ServiceDelivery			*+Structure*	See SIRI Part 2 ***ServiceDelivery***
HEADER	*:::*	**1:1**	See ServiceDelivery	
Payload	***GeneralMessageDelivery***	**1:***	*+Structure*	See ***GeneralMessageDelivery*** element.

12.6.3 GeneralMessageDelivery — Element

A ***GeneralMessageDelivery*** (see Table 109 below) is made up of zero, one or many ***InfoMessage*** elements, each representing a distinct informative message. The format of the message is set by implementation and by request.

Each ***InfoMessage*** included in the response has its own identifier, issued by the producer: this can be used to revoke previous messages using an ***InfoMessageCancellation***.

Table 109 — GeneralMessageDelivery — Attributes

GeneralMessageDelivery			*+Structure*	Message content & changes to messages.
Attributes	***version***	**1:1**	*VersionString*	Version identifier of ***GeneralMessage*** Service. Fixed e.g. '1.1a'.
LEADER	*:::*	**1:1**	xxx*ServiceDelivery*	See SIRI Part 2 xxx***ServiceDelivery***.
Payload	***InfoMessage***	0:*	*+Structure*	An informative message. See ***InfoMessage*** element.
	InfoMessage-Cancellation	0:*	*+Structure*	Reference to a previously communicated ***InfoMessage*** - which should not now be used.

12.6.4 InfoMessage — Element

Each ***InfoMessage*** describes an informational message (see Table 105 below).

If a new ***InfoMessage*** is sent with the same ***InfoMessageIdentifier*** as a previous ***InfoMessage***, it overwrites the content of the old message. This allows the messages to be updated. It is not possible to update (overwrite) a previously deleted message.

Table 110 — InfoMessage — Attributes

InfoMessage			*+Structure*	An informative message.
attribute	***formatRef***	0:1	*→FormatCode*	Identifies the format of the Content. If absent, free text.
log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which vehicle data was recorded.
Identity	***ItemIdentifier***	0:1	*ItemIdentifier*	Unique identifier of Item. Can be used to reference it.
Identity	***InfoMessage-Identifier***	**1:1**	*Identifier*	***InfoMessage*** identifier.
	InfoMessageVersion	0:1	*xsd:positive-Integer*	Version of ***InfoMessage***. If absent v.1.
	InfoChannelRef	0:1	*→InfoChannel*	***InfoChannel*** to which message belongs.
Currency	***ValidUntilTime***	0:1	*xsd:dateTime*	Time until which message is valid. If absent open ended.
Situation	***SituationRef***	0:*	*→SituationCode*	Reference to a Situation associated with the element.
Message	***Content***	**1:1**	*xsd:anyType*	Message Content. Format is specified by ***FormatRef***. May be a simple string or an embedded XML document
any	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

12.6.5 InfoMessageCancellation — Element

If an ***InfoMessage*** loses its validity before normal expiration according to the ***ValidUntilTime***, the Producer can inform the Consumer with the transmission of an ***InfoMessageCancellation*** message (see Table 111 below). Each ***InfoMessageCancellation*** revokes a previous informational message.

Table 111 — InfoMessageCancellation — Attributes

InfoMessageCancellation			*+Structure*	Describes the progress of a VEHICLE along its route.
log	***RecordedAtTime***	**1:1**	*xsd:dateTime*	Time at which vehicle data was recorded.
Identity	***ItemRef***	0:1	*→ItemIdentifier*	Reference to a previous ***InfoMessage*** which this item cancels.
Identity	***InfoMessage-Identifier***	**1:1**	*Identifier*	***InfoMessage*** identifier.
	InfoChannelRef	0:1	*→Info-ChannelCode*	***InfoChannel*** to which message belongs.
xsd	***Extensions***	0:1	*xsd:any**	Placeholder for user extensions.

12.6.6 GeneralMessageDelivery — Example

The following is an example of a ***GeneralMessageDelivery*** using a simple message string.

```
<ServiceDelivery>
    <!--========HEADER============================================== -->
    <ResponseTimestamp>2004-12-17T09:30:46-05:00</ResponseTimestamp>
    <ProducerRef>KUBRICK</ProducerRef>
    <Status>true</Status>
    <MoreData>false</MoreData>
    <!--========PAYLOAD============================================= -->
    <GeneralMessageDelivery version="1.0">
        <ResponseTimestamp>2001-12-17T09:30:47.0Z</ResponseTimestamp>
        <RequestMessageRef>12345</RequestMessageRef>
```

```
            <Status>true</Status>
        <!--=======Functional SERVICE PAYLOAD========================== -->
            <InfoMessage FormatRef=”String”>
                <RecordedAtTime>2001-12-17T09:30:47.0Z</RecordedAtTime>
                <InfoMessageIdentifier>00034567</InfoMessageIdentifier>
                <InfoMessageVersion>2</InfoMessageVersion>
                <InfoChannelRef>WARNING</InfoChannelRef>
                <ValidUntilTime>2001-12-17T09:30:47.0Z</ValidUntilTime>
                <Content>Beware the Ides of March</Content>
            </InfoMessage>
            <InfoMessageCancellation>
                <InfoMessageIdentifier>00034564</InfoMessageIdentifier>
                <InfoChannelRef>WARNING</InfoChannelRef>
            </InfoMessageCancellation>
        </GeneralMessageDelivery>
    </ServiceDelivery>
```

The following is an example of a ***GeneralMessageDelivery*** using an embedded message format.

```
<ServiceDelivery>
        <ResponseTimestamp>2005-12-16T16:30:46-05:00</ResponseTimestamp>
        <ProducerRef>KUBRICK</ProducerRef>
        <GeneralMessageDelivery version="1.0j">
            <ResponseTimestamp>2005-12-16T16:30:47.0Z</ResponseTimestamp>
            <GeneralMessage formatRef="Xtis version=1.0">
                <RecordedAtTime>2005-12-16T10:58:11Z </RecordedAtTime>
                <InfoMessageIdentifier>00034567</InfoMessageIdentifier>
                <ValidUntilTime>2005-12-18T23:59:59Z </ValidUntilTime>
                <Content>
                    <Xtis version="1.0">
                        <Incidents>
                            <PtIncident>
                                <CreationTime>2005-12-16T10:58:11Z</CreationTime>
                                <IncidentNumber>00034567</IncidentNumber>
                                    <Verification>verified</Verification>
                                <Progress>open</Progress>
                                <ValidityPeriod>
                                    <StartTime>2005-12-16T10:49:00Z</StartTime>
                                    <EndTime>2005-12-18T23:59:59Z</EndTime>
                                </ValidityPeriod>
                                <EquipmentReason>signalFailure</EquipmentReason>
                                <Severity>normal</Severity>
                                <Summary>Signal failure at Bakerloo</Summary>
                                <Affects>
                                    <Networks>
                                        <AffectedNetwork>
                                            <AffectedLine>
                                                <LineRef>01BAK</LineRef>
                                                <Direction>
                                                    <DirectionRef>R</DirectionRef>
                                                </Direction>
                                            </AffectedLine>
                                        </AffectedNetwork>
                                    </Networks>
                                    <StopPoints>
                                        <AffectedStopPoint>
                                            <StopPointRef>490G00000011</StopPointRef>
                                            <StopPointName>Baker Street</StopPointName>
                                            <ZoneName>London</ZoneName>
                                        </AffectedStopPoint>
                                    </StopPoints>
                                </Affects>
```

```
                        </PtIncident>
                    </Incidents>
                </Xtis>
            </Content>
        </GeneralMessage>
    </GeneralMessageDelivery>
</ServiceDelivery>
```

Bibliography

[1] EN 12896, *Road transport and traffic telematics - Public transport - Reference data model*

[2] EN 28701, *Intelligent transport systems - Public transport - Identification of Fixed Objects in Public Transport (IFOPT)*

[3] CEN/TS 16614-1, *Public transport - Network and Timetable Exchange (NeTEx) - Part 1: Public transport network topology exchange format*

[4] CEN/TS 16614-2, *Public transport - Network and Timetable Exchange (NeTEx) - Part 2: Public transport scheduled timetables exchange format*

This page deliberately left blank

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email bsmusales@bsigroup.com.

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Copyright

Useful Contacts:

Customer Services
Tel: +44 845 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 845 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

bsi.

...making excellence a habit.™