Technical Specification

ISO/TS 21219-13

First edition
2025-01

Intelligent transport systems — Traffic and travel information via transport protocol experts group, generation 2 (TPEG2) —

Part 13: Public transport information service (TPEG2-PTS)

Systèmes intelligents de transport — Informations sur le trafic et le tourisme via le groupe expert du protocole de transport, génération 2 (TPEG2) —

Partie 13: Service d'informations sur les transports publics (TPEG2-PTS)

Reference number
ISO/TS 21219-13:2025(en)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*.

A list of all parts in the ISO 21219 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

0.1 History

TPEG technology was originally proposed by the European Broadcasting Union (EBU) Broadcast Management Committee, who established the B/TPEG project group in the autumn of 1997 with a brief to develop, as soon as possible, a new protocol for broadcasting traffic and travel-related information in the multimedia environment. TPEG technology, its applications and service features were designed to enable travel-related messages to be coded, decoded, filtered and understood by humans (visually and/or audibly in the user's language) and by agent systems. Originally, a byte-oriented data stream format, which can be carried on almost any digital bearer with an appropriate adaptation layer, was developed. Hierarchically structured TPEG messages from service providers to end-users were designed to transfer information from the service provider database to an end-user's equipment.

One year later, in December 1998, the B/TPEG group produced its first EBU specifications. Two documents were released. Part 2 (TPEG-SSF, which became ISO/TS 18234-2) described the syntax, semantics and framing structure, which was used for all TPEG applications. Meanwhile, Part 4 (TPEG-RTM, which became ISO/TS 18234-4) described the first application for road traffic messages.

Subsequently, in March 1999, CEN/TC 278, in conjunction with ISO/TC 204, established a group comprising members of the former EBU B/TPEG and this working group continued development work. Further parts were developed to make the initial set of four parts, enabling the implementation of a consistent service. Part 3 (TPEG-SNI, ISO/TS 18234-3) described the service and network information (SNI) application used by all service implementations to ensure appropriate referencing from one service source to another.

Part 1 (TPEG-INV, ISO/TS 18234-1) completed the series by describing the other parts and their relationship. It also contained the application IDs used within the other parts. Additionally, Part 5, the public transport information application (TPEG-PTI, ISO/TS 18234-5), was developed. The so-called TPEG-LOC location referencing method, which enabled both map-based TPEG-decoders and non-map-based ones to deliver either map-based location referencing or human readable text information, was issued as ISO/TS 18234-6 to be used in association with the other applications of parts of the ISO 18234 series to provide location referencing.

The ISO 18234 series has become known as TPEG Generation 1.

0.2 TPEG Generation 2

When the Traveller Information Services Association (TISA), derived from former forums, was inaugurated in December 2007, TPEG development was taken over by TISA and continued in the TPEG applications working group.

It was about this time that the (then) new Unified Modelling Language (UML) was seen as having major advantages for the development of new TPEG applications in communities who would not necessarily have the binary physical format skills required to extend the original TPEG TS work. It was also realized that the XML format for TPEG described within the ISO 24530 series (now superseded) had a greater significance than previously foreseen, especially in the content-generation segment, and that keeping two physical formats synchronized, in different standards series, would be rather difficult.

As a result, TISA set about the development of a new TPEG structure that would be UML-based. This has subsequently become known as TPEG Generation 2.

TPEG2 is embodied in the ISO 21219 series and it comprises many parts that cover the introduction, rules, toolkit and application components. TPEG2 is built around UML modelling and has a core of rules that contain the modelling strategy covered in ISO 21219-2, ISO 21219-3 and ISO 21219-4 and the conversion to two current physical formats: binary (Annex A) and XML (Annex B); others can be added in the future. TISA uses an automated tool to convert from the agreed UML model XMI file directly into an MS Word document file that forms the annex for each physical format.

TPEG2 has a three-container conceptual structure: message management (ISO 21219-6), application (several parts) and location referencing (ISO 21219-7). This structure has flexible capability and can accommodate

many differing use cases that have been proposed within the TTI sector and more broadly for hierarchical message content.

TPEG2 also has many location referencing options as required by the service provider community, any of which may be delivered by vectoring data included in the location referencing container.

The following classification provides a helpful grouping of the different TPEG2 parts according to their intended purpose. Note that the list below may be incomplete, as it is possible that new TPEG2 parts will be introduced after the publication of this document.

— Toolkit parts: TPEG2-INV (ISO 21219-1), TPEG2-UML (ISO 21219-2), TPEG2-UBCR (ISO 21219-3), TPEG2-UXCR (ISO 21219-4), TPEG2-SFW (ISO 21219-5), TPEG2-MMC (ISO 21219-6), TPEG2-LRC (ISO 21219-7).

— Special applications: TPEG2-SNI (ISO 21219-9), TPEG2-CAI (ISO 21219-10), TPEG2-LTE (ISO/TS 21219-24).

— Location referencing: TPEG2-OLR (ISO/TS 21219-22), TPEG2-GLR (ISO 21219-21), TPEG2-TLR (ISO 17572-2), TPEG2-DLR (ISO 17572-3).

— Applications: TPEG2-PTS (ISO 21219-13 – this document), TPEG2-PKI (ISO 21219-14), TPEG2-TEC (ISO 21219-15), TPEG2-FPI (ISO 21219-16), TPEG2-SPI (ISO 21219-17), TPEG2-TFP (ISO 21219-18), TPEG2-WEA (ISO 21219-19), TPEG2-RMR (ISO/TS 21219-23), TPEG2-EMI (ISO 21219-25), TPEG2-VLI (ISO/TS 21219-26).

TPEG2 has been developed to be broadly (but not totally) backward compatible with TPEG1 to assist in transitions from earlier implementations, while not hindering the TPEG2 innovative approach and being able to support many new features, such as dealing with applications with both long-term, unchanging content and highly dynamic content, such as parking information.

This document is based on the TISA specification technical/editorial version reference:

SP19008/1.0/001.

Intelligent transport systems — Traffic and travel information via transport protocol experts group, generation 2 (TPEG2) —

Part 13: Public transport information service (TPEG2-PTS)

1 Scope

This document describes the "public transport information service" (PTS) application, which is intended to cover all modes of public (i.e. collective) transport, both for inter-urban and intra-urban travel. The PTS application is designed to allow the efficient and language-independent delivery of public transport information directly from a service provider to end-users.

The PTS application design is based on three main use cases.

— Provision of alert information: an alert is a warning that indicates an emergency situation. This case is specifically relevant for broadcast/push mode, for major deviations or disruptions which are relevant for a large number of travellers. A dedicated alert request is also defined and can be used if a backchannel is available.

— Timetable information, both scheduled and real time: this information is in some cases relevant for broadcast, e.g. in case of large events for the transport modalities to/from the event site. A dedicated timetable request is also defined and can be used if a backchannel is available.

— Individual requests for trip information (backchannel is required).

The PTS application focuses on providing core information regarding public transport in order to ensure the compactness of the TPEG application. Specific information as provided in typical public transport apps (e.g. fare information) is not in the scope of this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 21219-1, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 1: Introduction, numbering and versions (TPEG2-INV)*

ISO 21219-7, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 7: Location referencing container (TPEG2-LRC)*

ISO 21219-9, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 9: Service and network information (TPEG2-SNI)*

ISO 21219-14, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 14: Parking information (TPEG2-PKI)*

ISO 21219-15, *Intelligent transport systems — Traffic and travel information (TTI) via transport protocol experts group, generation 2 (TPEG2) — Part 15: Traffic event compact (TPEG2-TEC)*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
AccessFeature
facility providing access to a StopPlace, StopPoint or vehicle

EXAMPLE Elevator, stairs, ramp.

Note 1 to entry: This description is close to the Open API for Distributed Journey Planning term "AccessFeatureType", specified in CEN/TS 17118:2017, Table 119. The meaning is similar, but harmonized to the PTS (public transport information service) application context.

3.2
Destination
place to where the user is heading

Note 1 to entry: In the PTS (public transport information service) application context, this can be StopPlaces or StopPoints only. PTS additionally uses Destination to describe the End of a VehicleJourney as specified by CEN/TS 17118:2017, Table 92.

3.3
Line
aggregation of similar VehicleJourneys which are published under the same name

EXAMPLE Bus line 100, or airport shuttle.

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "Line", specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

3.4
ModeOfTransport
type of a VehicleJourney or Line

EXAMPLE Bus service, railway service, air service.

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "Mode", specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

3.5
CallAtStop
meeting of a VehicleJourney with a specific scheduled StopPoint

[SOURCE: CEN/TS 17118:2017, 3.1.123]

3.6
operator
company providing public transport services

[SOURCE: CEN/TS 17118:2017, 3.1.71]

3.7
Origin
Place from where the user wants to start

Note 1 to entry: In the PTS (public transport information service) application context, this can be StopPlaces or StopPoints only. PTS additionally uses Origin to describe the Start of a VehicleJourney, as specified by CEN/TS 17118:2017, Table 92.

3.8
PublishedLineName
name which is used for a Line in public

EXAMPLE Bus line 100, or airport shuttle.

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "PublishedLineName", specified by CEN/TS 17118:2017. The meaning is similar, but harmonized to the PTS application context.

3.9
Route
ordered list of located points defining one single path through the road (or rail) network

[SOURCE: EN 12896-1:2016]

3.10
StopEvent
departure or arrival event or both

[SOURCE: CEN/TS 11718:2017, 3.1.115]

3.11
StopPlace
one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names

EXAMPLE Station, airport, harbour.

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "StopPlace", specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

3.12
StopPoint
location with an identifier and name where passengers can board or alight from vehicles

EXAMPLE Platform, gate.

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "StopPoint", specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

3.13
Trip
whole journey from a passengers Origin to passenger Destination in one or more TripLegs

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "Trip", specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

3.14
TripLeg
single stage of a Trip that is made without change of ModeOfTransport or VehicleJourney

Note 1 to entry: This PTS description is close to the Open API for Distributed Journey Planning term "TripLeg", specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

3.15
VehicleJourney
description of a journey of a vehicle from its Origin to its Destination

Note 1 to entry: This PTS description is close to the OJP term VehicleJourney specified by CEN/TS 17118. The meaning is similar, but harmonized to the PTS application context.

4 Abbreviated terms

For the purposes of this document, the abbreviated terms in ISO 21219-1, ISO 21219-9, ISO 21219-14, ISO 21219-15 and the following apply.

OJP Open API for distributed Journey Planning

5 Application specific constraints

5.1 Application identification

The word “application” is used in the TPEG specifications to describe specific subsets of the TPEG structure. An application defines a limited vocabulary for a certain type of messages, e.g. parking information or road traffic information. Each TPEG application is assigned a unique number, called the AID. An AID is defined in ISO 21219-1 whenever a new application is developed.

The AID number is used within the TPEG2-SNI application[3] to indicate how to process TPEG content. It also facilitates the routing of information to the appropriate application decoder.

5.2 Version number signalling

Version numbering is used to track the separate versions of an application through its development and deployment. The differences between these versions could have an impact on client devices.

The version numbering principle is defined in ISO 21219-1.

Table 1 shows the current version numbers for signalling PTS within the SNI application.

Table 1 — Current version numbers for signalling of PTS

Major version number	1
Minor version number	0

5.3 Ordered components

TPEG2-PTS requires a fixed order of TPEG components. The order for the PTS message component is shown in Figure 1; the first component shall be the Message Management Container (MMC). This shall be the only component if the message is a cancellation message. Otherwise, the MMC component shall be followed by the one or more Application Data Container component(s) which includes the application-specific information.

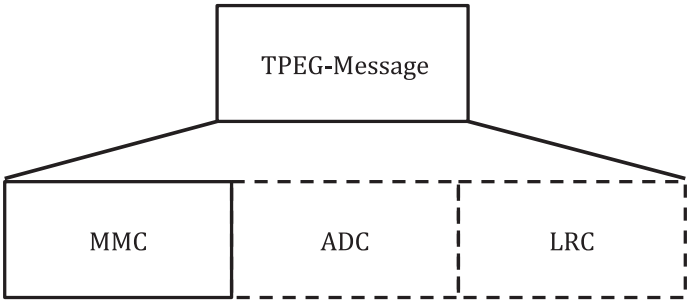

Figure 1 — Composition of TPEG messages

5.4 Extendibility

The requirement of a fixed component order does not affect the extension of TPEG2-PTS. Future application extensions may insert new components or may replace existing components by new ones without losing backward compatibility, i.e. a TPEG2-PTS decoder shall be able to detect and skip unknown components.

5.5 TPEG Service Component Frame

TPEG2-PTS shall make use of the "Service Component Frame with dataCRC and messageCount" according to ISO 21219-5.

6 PTS Structure

The structure of a PTS message is shown in Figure 2. This structure conforms to the UML modelling rules defined in ISO 21219-2. The binary format and XML format of the TPEG2-PTS application for use in transmission shall be in accordance with Annexes A and B, respectively. Examples of TPEG2-PTS use cases are shown in Annex C.

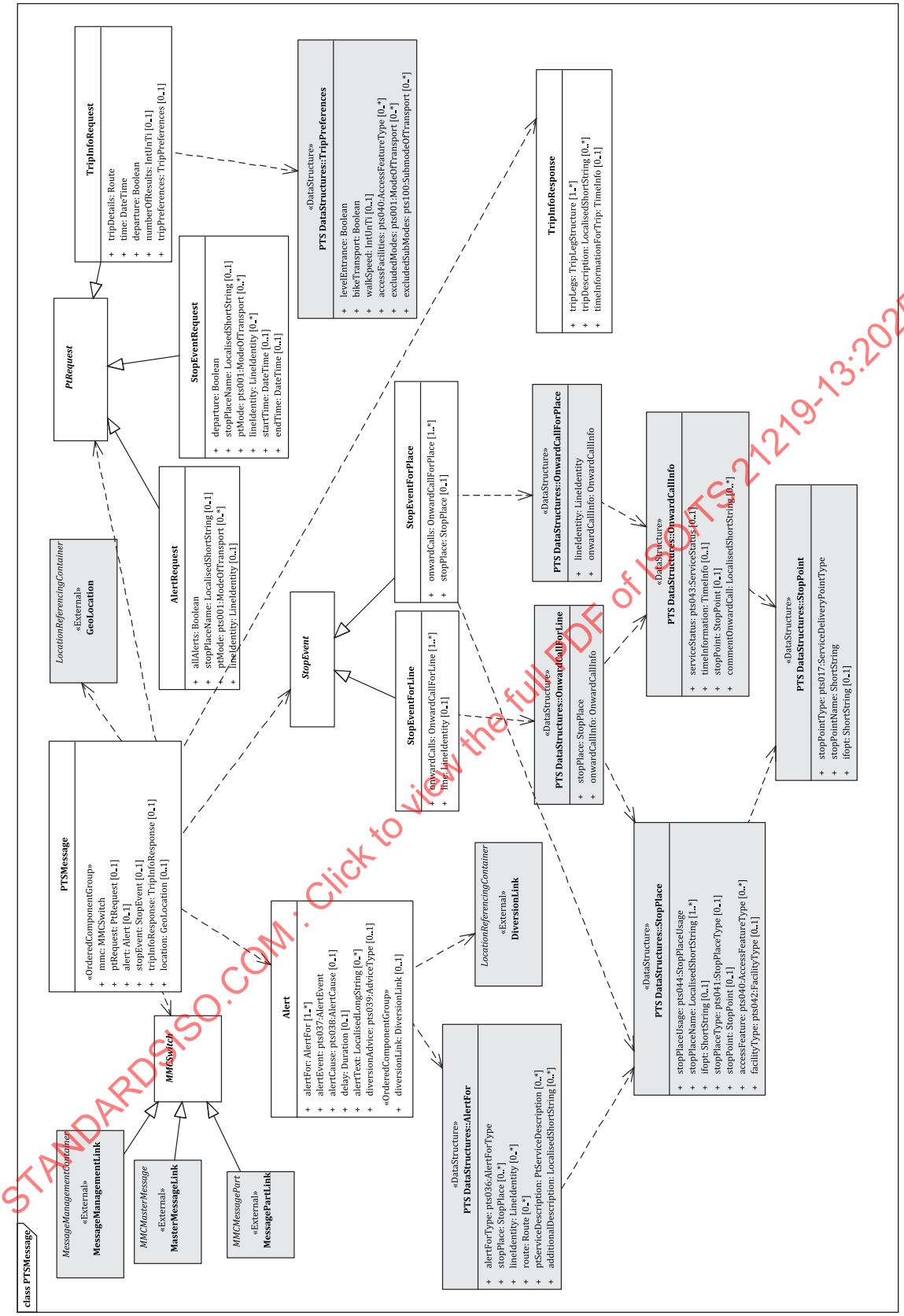

Figure 2 — PTS message structure

The structure of the TripLegStructure is shown in Figure 3.

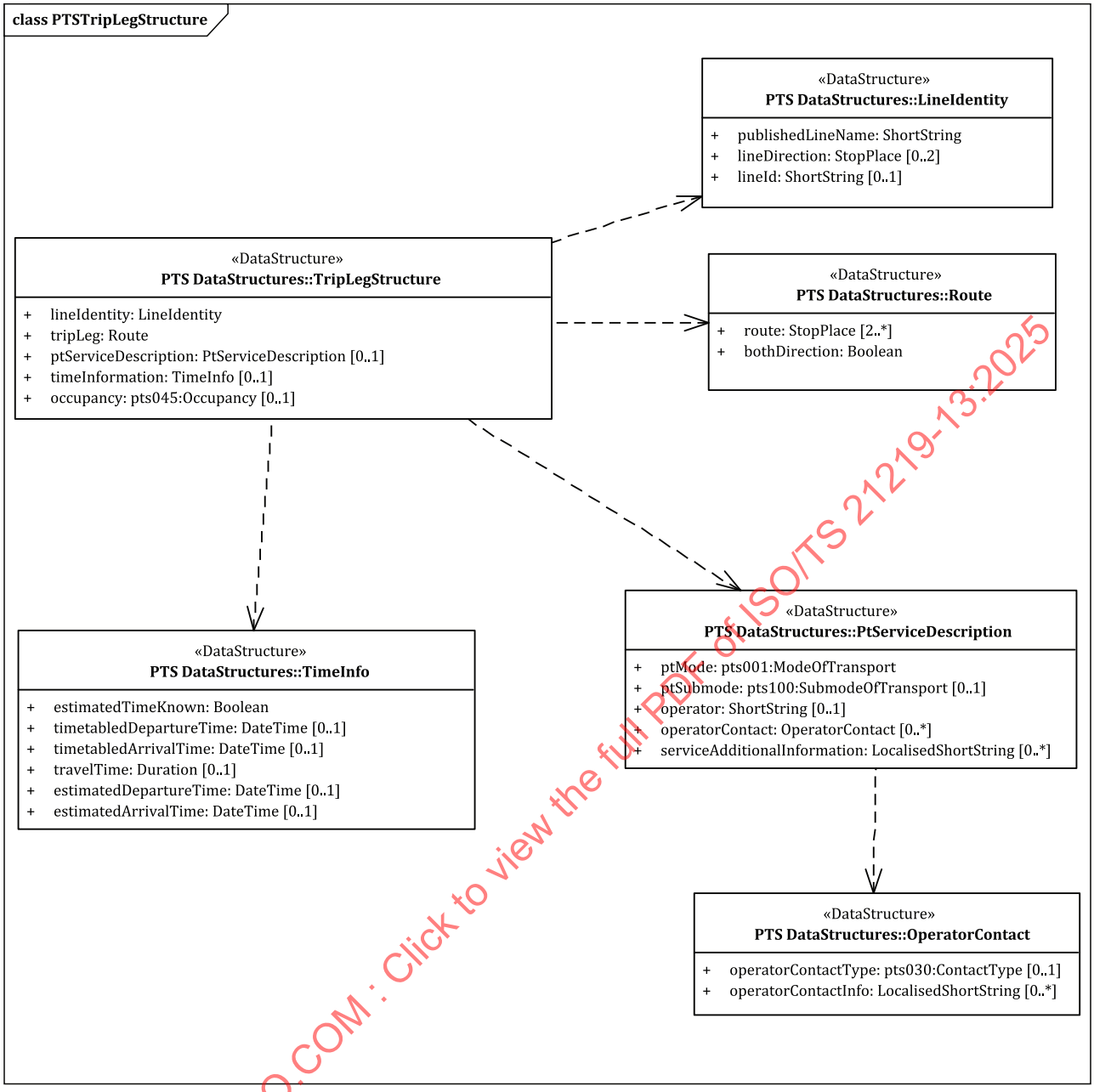

Figure 3 — TripLegStructure component structure

7 PTS message components

7.1 PTSMessage

Table 2 defines the PTSMessage component, i.e. the overall PTS message container.

Table 2 — PTSMessage

Name	Type	Multiplicity	Description
Ordered components			
mmc	MMCSwitch	1	n.a.
ptRequest	PtRequest	0..1	Component to transmit a request for public transport information.
alert	Alert	0..1	Component to transmit alert information.
stopEvent	StopEvent	0..1	Component to transmit information for a departure/ arrival events (timetable) at a stopPlace or for a Line.
tripInfoResponse	TripInfoResponse	0..1	Component to transmit the response to a TripInfoRequest.
location	GeoLocation	0..1	Map-related location referencing information for this PTS message.

7.2 MMCSwitch

The MMCSwitch component is an abstract component, allowing the flexible use of monolithic or multi-part message management.

7.3 MessageManagementLink

The MessageManagementLink is a placeholder for the MessageManagementContainer as defined by ISO 21219-6. It assigns the PTS application-specific local component ID for the MMC container (see Annex A).

7.4 MasterMessageLink

The MMCMasterLink is a placeholder for the Master-Message MMC for multi-part message management, as defined by ISO 21219-6. It assigns the PTS application-specific local component ID for the MMC container (see Annex A).

7.5 MessagePartLink

The MMCPartLink is a placeholder for the external partial-Message MMC (MMCMessagePart) for multi-part message management, as defined by ISO 21219-6. It assigns the PTS application-specific local component ID for the MMC container (see Annex A).

7.6 GeoLocation

The GeoLocation component is a placeholder for the LocationReferencingContainer (LRC). It assigns the PTS application-specific local component ID for the LRC container. All component IDs within the LRC container shall be taken from the LRC toolkit specified by ISO 21219-7.

The LocationReferenceLink component is a placeholder for the LocationReferencingContainer (LRC). It assigns the PTS application-specific local component ID for the LRC container. All component IDs within the LRC container shall be taken from the LRC toolkit specified by ISO 21219-7.

The GeoLocation describes the location for which the information is intended. e.g. airport, railway station, area, city, line.

7.7 PtRequest

Component to transmit a request for public transport information. This component can contain one of the following: AlertRequest, StopEventRequest or TripInfoRequest.

7.8 AlertRequest

Table 3 defines the AlertRequest component, i.e. the component to transmit a request for one or more alerts.

Table 3 — AlertRequest

Name	Type	Multiplicity	Description
allAlerts	Boolean	1	True if an AlertRequest is issued for all available alerts; false otherwise. If false, the remaining attributes specify the AlertRequest.
stopPlaceName	LocalisedShortString	0..1	Well-known name of the StopPlace for which an AlertRequest is issued.
ptMode	pts001:ModeOfTransport	0..*	List of the public transportation modes for which an AlertRequest is issued.
lineIdentity	LineIdentity	0..1	Description of the Line for which an AlertRequest is issued.

7.9 StopEventRequest

The StopEventRequest component is used to transmit a request for stop-centric arrivals/departures (a timetable or departure board):

— either for one or more Lines at a stopPlace, or

— for all StopPlaces on a Line.

Table 4 defines the StopEventRequest component.

Table 4 — StopEventRequest

Name	Type	Multiplicity	Description
departure	Boolean	1	True if a StopEventRequest is issued for departures; false if a StopEventRequest is issued for arrivals.
stopPlaceName	LocalisedShortString	0..1	In the case of a StopEventRequest for a StopPlace, describes the well-known name of the StopPlace.
ptMode	pts001:ModeOfTransport	0..*	In the case of a StopEventRequest for a StopPlace, lists the modes of transport to be considered for the request.
lineIdentity	LineIdentity	0..*	In the case of a StopEventRequest for a Line, description of the Line to be considered for the request.
startTime	DateTime	0..1	Start of the timeframe for which the request is issued.
endTime	DateTime	0..1	Stop of the timeframe for which the request is issued.

7.10 TripInfoRequest

Table 5 defines the TripInfoRequest component, i.e. the component to transmit a request for trip information.

Table 5 — TripInfoRequest

Name	Type	Multiplicity	Description
tripDetails	Route	1	Description of the origin and destination of a trip with potentially one or more vias.
time	DateTime	1	Contains either Departure time at origin or Arrival time at destination.
departure	Boolean	1	True to indicate that time attribute contains Departure time at origin, false to indicate that time attribute contains Arrival time at destination.
numberOfResults	IntUnTi	0..1	Attribute to control the number of trip results before/after a point in time. May not be used when departure time at origin AND arrival time at destination are set.
tripPreferences	TripPreferences	0..1	User preferences to be considered in trip calculation.

7.11 Alert

Table 6 defines the Alert component, i.e. the component to transmit alert information, typically to be used for the use case where major alerts are being sent by broadcast.

Table 6 — Alert

Name	Type	Multiplicity	Description
alertFor	AlertFor	1..*	Describes the StopPlace, Line, Route, Area, etc. for which the alert is issued.
alertEvent	pts037:AlertEvent	1	Information on the actual event (e.g. closure of a line, long delays of a line, cancellation of a stop).
alertCause	pts038:AlertCause	0..1	Information on the cause of the event.
delay	Duration	0..1	In the case of a delay, this component can be used to inform about its duration (in min).
alertText	LocalisedLongString	0..*	Free text option to provide additional description of the alert.
diversionAdvice	pts039:AdviceType	0..1	Information on a potential diversion, e.g. to use an alternative route.
Ordered components			
diversionLink	DiversionLink	0..1	Map-related location referencing information for a potential diversion.

7.12 DiversionLink

The DiversionLink component contains map-related location referencing information for a potential diversion.

7.13 StopEvent

The StopEvent component is used to transmit information for departure/arrival events (departure board or timetable):

— either for one or more Lines at a stopPlace, or

— for one or more StopPlaces on a Line.

7.14 StopEventForPlace

Table 7 defines the StopEventForPlace component, i.e. the component to transmit information for departure/arrival events departure board or timetable, for one or more Lines at a StopPlace.

Table 7 — StopEventForPlace

Name	Type	Multiplicity	Description
callAtStops	CallAtStopForPlace	1..*	Provides information on departure/arrival events for each relevant Line (a stop timetable).
stopPlace	StopPlace	0..1	Describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.

7.15 StopEventForLine

Table 8 defines the StopEventForLine component, i.e. the component to transmit information for departure/arrival events (departure board or timetable) for a Line or VehicleJourney.

Table 8 — StopEventForLine

Name	Type	Multiplicity	Description
callAtStops	CallAtStopForLine	1..*	Provides information on departure/arrival events for each relevant StopPlace (along the Line).
line	LineIdentity	0..1	Description of a Line.

7.16 TripInfoResponse

Table 9 defines the TripInfoResponse component, i.e. the component to transmit a response to a TripInfoRequest.

Table 9 — TripInfoResponse

Name	Type	Multiplicity	Description
tripLegs	TripLegStructure	1..*	Describes one or more TripLegs for the trip. A TripLeg being a single stage of a Trip that is made without change of ModeOfTransport or Line (i.e. between each Transfer).
tripDescription	LocalisedShortString	0..*	Descriptive text for a Trip.
timeInformationForTrip	TimeInfo	0..1	Describes scheduled and/or actual departure and/or arrival times for a Trip.

8 PTS Datatypes

8.1 AlertFor

Table 10 defines the AlertFor datatype that describes the entity (StopPlace, Line, Route, PT service, Area etc.) for which the alert is issued (one such entity per Alert message).

Table 10 — AlertFor

Name	Type	Multiplicity	Description
alertForType	pts036:AlertForType	1	Information on the type of entity for which the alert is issued.
stopPlace	StopPlace	0..*	Describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.
lineIdentity	LineIdentity	0..*	Description of a Line.
route	Route	0..*	An ordered list of located StopPlaces defining one single path through the road or rail network.
ptServiceDescription	PtServiceDescription	0..*	Information on a public transport service.
additionalDescription	LocalisedShortString	0..*	Textual description of an area.

8.2 LineIdentity

Table 11 defines the LineIdentity datatype that contains the description of a Line.

Table 11 — LineIdentity

Name	Type	Multiplicity	Description
publishedLineName	ShortString	1	Line name or service description as known to the public.
lineDirection	StopPlace	0..2	StartPoint and Endpoint of a Line.
lineId	ShortString	0..1	Identifier of a Line.

8.3 CallAtStopForLine

Table 12 defines the CallAtStopForLine datatype.

Table 12 — CallAtStopForLine

Name	Type	Multiplicity	Description
stopPlace	StopPlace	1	Describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.
callAtStopInfo	CallAtStopInfo	1	Provides information on departure/arrival events for one certain StopPlace.

8.4 CallAtStopForPlace

Table 13 defines the CallAtStopForPlace datatype.

Table 13 — CallAtStopForPlace

Name	Type	Multiplicity	Description
lineIdentity	LineIdentity	1	Describes a Line.
callAtStopInfo	CallAtStopInfo	1	Provides information on departure/arrival events for one certain Line.

8.5 CallAtStopInfo

Table 14 defines the CallAtStopInfo datatype.

Table 14 — CallAtStopInfo

Name	Type	Multiplicity	Description
serviceStatus	pts043:ServiceStatus	0..1	Describes the status of the service.
timeInformation	TimeInfo	0..1	Describes scheduled or estimated departure times or both and the same applies for arrival times.
stopPoint	StopPoint	0..1	Describes a physical area within a StopPlace (e.g. a platform).
commentCallAtStop	LocalisedShortString	0..*	Additional descriptive text.

8.6 OperatorContact

Table 15 defines the OperatorContact datatype.

Table 15 — OperatorContact

Name	Type	Multiplicity	Description
operatorContactType	pts030:ContactType	0..1	Information on the type of possibility to contact the operator.
operatorContactInfo	LocalisedShortString	0..*	Describes the contact information for the operator.

8.7 PtServiceDescription

Table 16 defines the PtServiceDescription datatype that contains the description of a public transport service.

Table 16 — PtServiceDescription

Name	Type	Multiplicity	Description
ptMode	pts001:ModeOfTransport	1	Defines the public transport mode.
ptSubmode	pts100:SubmodeOfTransport	0..1	Defines the public transport submode.
operator	ShortString	0..1	Operator name as known to the public.
operatorContact	OperatorContact	0..*	Describes a contact for the operator.
serviceAdditionalInformation	LocalisedShortString	0..*	Additional descriptive information about the public transport service.

8.8 Route

Table 17 defines the Route datatype that describes an ordered list of located StopPlaces defining one single path through the road or rail network.

Table 17 — Route

Name	Type	Multiplicity	Description
route	StopPlace	2..*	Array containing the StopPlaces on the Route.
bothDirection	Boolean	1	Only relevant for Alert: true when an Alert is issued for both directions of the route.

8.9 StopPlace

Table 18 defines the StopPlace datatype that describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.

Table 18 — StopPlace

Name	Type	Multiplicity	Description
stopPlaceUsage	pts044:StopPlaceUsage	1	Identifies the "role" a StopPlace has: — in a Trip (e.g. origin, destination, intermediate); or — for an Alert (e.g. temporarily non-stopping); or — on a Line (e.g. temporary stop place, first route point).
stopPlaceName	LocalisedShortString	1..*	Well-known name of the StopPlace.
stopPlaceRef	ShortString	0..1	Reference key for the StopPlace.
stopPlaceType	pts041:StopPlaceType	0..1	Information on the type of StopPlace (e.g. railway station, bus station).
stopPoint	StopPoint	0..1	Describes a physical area within a StopPlace (e.g. a platform).
accessFeature	pts040:AccessFeatureType	0..*	Describes accessibility features associated with this StopPlace.
facilityType	pts042:FacilityType	0..1	Named amenity available to the public at this StopPlace.

8.10 StopPoint

Table 19 defines the StopPoint datatype that describes a location with identifier and name where passengers can board or alight from vehicles, e.g. platform, gate.

Table 19 — StopPoint

Name	Type	Multiplicity	Description
stopPointType	pts017:ServiceDeliveryPointType	1	Information on the type of StopPoint (e.g. gate, platform).
stopPointName	ShortString	1	Well-known name of the StopPoint.
stopPointRef	ShortString	0..1	Reference key for the StopPoint.

8.11 TimeInfo

Table 20 defines the TimeInfo datatype that contains the description of scheduled or actual departure time or both and the same applies to arrival times and to travel times for various purposes.

Table 20 — TimeInfo

Name	Type	Multiplicity	Description
estimatedTimeKnown	Boolean	1	True if an expected/estimated time is known; false otherwise.
timetabledDepartureTime	DateTime	0..1	Departure time according to planned timetable.
timetabledArrivalTime	DateTime	0..1	Arrival time according to planned timetable.
travelTime	Duration	0..1	Travel time for a trip (in case of TripInfo); travel time for a TripLeg (in case of TripLegStructure).
estimatedDepartureTime	DateTime	0..1	Expected/estimated departure time at the StopPoint.
estimatedArrivalTime	DateTime	0..1	Expected/estimated arrival time at the StopPoint.

8.12 TripLegStructure

Table 21 defines the TripLegStructure datatype that describes a TripLeg.

Table 21 — TripLegStructure

Name	Type	Multiplicity	Description
lineIdentity	LineIdentity	1	Description of a Line.
tripLeg	Route	1	A single stage of a Trip that is made without change of ModeOfTransport or Line (i.e. between each Transfer).
ptServiceDescription	PtServiceDescription	0..1	Description of a public transport service.
timeInformation	TimeInfo	0..1	Description of scheduled or actual departure times or both at the starting point of the TripLeg and the same applies to the arrival times at the ending point of the TripLeg, as well as to the travel time.
occupancy	pts045:Occupancy	0..1	How full the vehicle is. If omitted, not known.

8.13 TripPreferences

Table 22 defines the TripPreferences datatype that provides information on user preferences to be considered in trip calculation.

Table 22 — TripPreferences

Name	Type	Multiplicity	Description
levelEntrance	Boolean	1	If set to true, the user needs vehicles with a level entrance between platform and vehicle, e.g. for wheelchair access. Lift-equipped vehicles or stationary lifts at the platform can be sufficient. Default is false.
bikeTransport	Boolean	1	If set to true, the user wants to carry a bike on public transport. Default is false.
walkSpeed	IntUnTi	0..1	Deviation from average walk speed in percent. 100 % percent means average speed. Values less than 100 % mean slower, greater than 100 % faster. Default is 100 %.
accessFacilities	pts040:AccessFeatureType	0..*	Specific accessibility/mobility requirements set to access paths and connection paths (with respect to the physical restrictions that apply to them).
excludedModes	pts001:ModeOfTransport	0..*	Public transport modes to be excluded in trip calculation.
excludedSubModes	pts100:SubmodeOfTransport	0..*	Public transport submodes to be excluded in trip calculation.

By using accessFacilities within the structure "TripPreferences" the TripInfoRequest can be restricted to a certain degree of accessibility. Thus the TripInfoResponse shall include or exclude facilities corresponding to the requested AccessFeature according to Table 23.

Table 23 — AccessFeatures

Request	**Response**								
	unknown	**single step**	**stairs**	**escalator**	**moving walkway**	**elevator**	**wheelchair ramp**	**mind the gap**	**undefined access feature type**
unknown	n/a	✓	✓	✓	✓	✓	✓	✓	✓
single step	×	n/a	×	×	✓	✓	✓	✓	×
stairs	✓	✓	n/a	✓	✓	✓	✓	✓	✓
escalator	×	✓	×	n/a	✓	✓	✓	✓	×
moving walkway	×	×	×	×	n/a	✓	✓	×	×
elevator	×	×	×	×	✓	n/a	✓	×	×
wheelchair ramp	×	×	×	×	✓	✓	n/a	×	×
mind the gap	×	×	×	×	✓	✓	✓	n/a	✓
undefined access feature type	✓	✓	✓	✓	✓	✓	✓	✓	n/a
Key ✓ include × exclude									

9 PTS Tables

9.1 pts001: ModeOfTransport

Table 24 enumerates the possible values for type pts001: ModeOfTransport.

Table 24 — pts001: ModeOfTransport

Code	Reference-English "word"	Comment	Example
0	unknown		
1	air service		
2	gondola cable car service		
3	chairlift service		
4	elevator service		
5	railway service		
6	urban railway service		
7	light railway service		
8	rack rail service		
9	funicular service		
10	bus service		
11	trolleybus service		
12	coach service		
13	taxi service		
14	rental vehicle		
15	water transport service		
16	cable drawn boat service		
255	undefined public transport service		

9.2 pts017: ServiceDeliveryPointType

Table 25 enumerates the possible values for type pts017: ServiceDeliveryPointType.

Table 25 — pts017: ServiceDeliveryPointType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	platform number		
2	terminal gate		
3	ferry berth		
4	harbour pier		
5	landing stage		
6	bus stop		
255	undefined service delivery point		

9.3 pts030: ContactType

Table 26 enumerates the possible values for type pts030: ContactType.

Table 26 — pts030: ContactType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	internet link		
2	telephone number	emergency number	
3	text calling number		
4	voice calling number		
5	data calling number		
6	price information		
7	ticketing information		
255	undefined contact type		

9.4 pts036: AlertForType

Table 27 enumerates the possible values for type pts036: AlertForType.

Table 27 — pts036: AlertForType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	stopplace		
2	line		
3	route		
4	public transport service		
5	operator		
6	city		
7	area		
8	stoppoint		
255	undefined type		

9.5 pts037: AlertEvent

Table 28 enumerates the possible values for type pts037: AlertEvent.

Table 28 — pts037: AlertEvent

Code	Reference-English "word"	Comment	Example
0	unknown		
1	closed		
2	delays		
3	long delays		
4	cancelled		
255	undefined alert event		

9.6 pts038: AlertCause

Table 29 enumerates the possible values for type pts038: AlertCause

Table 29 — pts038: AlertCause

Code	Reference-English "word"	Comment	Example
0	unknown		
1	security alert		
2	emergency services call		
3	police activity		
4	police order		
5	fire		
6	cable fire		
7	smoke detected on vehicle		
8	fire at the station		
9	fire run		
10	fire brigade order		
11	explosion		
12	explosion hazard		
13	bomb disposal		
14	emergency medical services		
15	emergency brake		
16	vandalism		
17	cable theft		
18	signal passed at danger		
19	station overrun		
20	passengers blocking doors		
21	defective security system		
22	overcrowded		
23	border control		
24	unattended bag		
25	telephoned threat		
26	suspect vehicle		
27	evacuation		
28	terrorist incident		
29	public disturbance		
30	technical problem		
31	vehicle failure		
32	service disruption		
33	door failure		
34	lighting failure		
35	points problem		
36	points failure		
37	signal problem		
38	signal failure		
39	overhead wire failure		

Table 29 *(continued)*

Code	Reference-English "word"	Comment	Example
40	level crossing failure		
41	traffic management system failure		
42	engine failure		
43	breakdown		
44	repair work		
45	construction work		
46	maintenance work		
47	power problem		
48	track circuit problem		
49	swing bridge failure		
50	escalator failure		
51	lift failure		
52	gangway problem		
53	defective vehicle		
54	broken rail		
55	poor rail conditions		
56	deicing work		
57	wheel problem		
58	route blockage		
59	congestion		
60	heavy traffic		
61	route diversion		
62	roadworks		
63	unscheduled construction work		
64	level crossing blocked		
65	sewerage maintenance		
66	road closed		
67	roadway damage		
68	bridge damage		
69	person on the line		
70	object on the line		
71	vehicle on the line		
72	animal on the line		
73	fallen tree on the line		
74	vegetation		
75	speed restrictions		
76	preceding vehicle		
77	accident		
78	near miss		
79	person hit by vehicle		
80	vehicle struck object		
81	vehicle struck animal		

Table 29 *(continued)*

Code	Reference-English "word"	Comment	Example
82	derailment		
83	collision		
84	level crossing accident		
85	poor weather		
86	fog		
87	heavy snowfall		
88	heavy rain		
89	strong winds		
90	ice		
91	hail		
92	high temperatures		
93	flooding		
94	low water level		
95	risk of flooding		
96	high water level		
97	fallen leaves		
98	fallen tree		
99	landslide		
100	risk of landslide		
101	drifting snow		
102	blizzard conditions		
103	storm damage		
104	lightning strike		
105	rough sea		
106	high tide		
107	low tide		
108	ice drift		
109	avalanches		
110	risk of avalanches		
111	flash floods		
112	mudslide		
113	rockfalls		
114	subsidence		
115	earthquake damage		
116	grass fire		
117	wildland fire		
118	ice on railway		
119	ice on carriages		
120	special event		
121	procession		
122	demonstration		
123	industrial action		
124	staff sickness		

Table 29 *(continued)*

Code	Reference-English "word"	Comment	Example
125	staff absence		
126	operator ceased trading		
127	previous disturbances		
128	vehicle blocking track		
129	foreign disturbances		
130	waiting for transfer passengers		
131	change in carriages		
132	train coupling		
133	boarding delay		
134	awaiting oncoming vehicle		
135	overtaking		
136	provision delay		
137	miscellaneous		
255	undefined alert cause		

9.7 pts039: AdviceType

Table 30 enumerates the possible values for type pts039: AdviceType.

Table 30 — pts039: AdviceType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	use replacement bus		
2	use replacement train		
3	use the alternative route		
4	go on foot		
5	please leave the station! danger!		
6	no means of travel		
7	use different stops		
8	use alternative stop		
9	do not leave vehicle! danger!		
10	take advice from announcements		
11	take advice from personnel		
12	obey advice from police		
255	undefined advice		

9.8 pts040: AccessFeatureType

Table 31 enumerates the possible values for type pts040: AccessFeatureType.

Table 31 — pts040: AccessFeatureType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	single step		
2	stairs		
3	escalator		
4	moving walkway		
5	elevator		
6	wheelchair ramp		
7	mind the gap		
8	tactile paving		
255	undefined access feature type		

9.9 pts041: StopPlaceType

Table 32 enumerates the possible values for type pts041: StopPlaceType.

Table 32 — pts041: StopPlaceType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	railway station		
2	underground station		
3	tram station		
4	bus station		
5	airport		
6	pier		
7	harbour port		
8	ferry stop		
9	light railway station		
10	cogwheel station		
11	funicular station		
12	ropeway station		
255	undefined stopplace type		

9.10 pts042: FacilityType

Table 33 enumerates the possible values for type pts042: FacilityType.

Table 33 — pts042: FacilityType

Code	Reference-English "word"	Comment	Example
0	unknown		
1	ticket shop		
2	ticket machine		
3	cab rank		
4	bike sharing available		
5	car sharing available		
6	seated		
7	sheltered		
8	security		
9	lighted		
10	toilets		
11	policy		
12	first aid		
13	sos point		
14	specific assistance		
15	unaccompanied minor assistance		
16	boarding assistance		
255	undefined facility		

9.11 pts043: ServiceStatus

Table 34 enumerates the possible values for type pts043: ServiceStatus.

Table 34 — pts043:ServiceStatus

Code	Reference-English "word"	Comment	Example
0	unknown		
1	delay		
2	minor delays		
3	major delays		
4	operation time extension		
5	on time		
6	disturbance rectified		
7	change of platform		
8	line cancellation		
9	trip cancellation		
10	boarding		
11	go to gate		
12	stop cancelled		
13	stop moved		
14	stop on demand		
15	additional stop		
16	substituted stop		

Table 34 *(continued)*

Code	Reference-English "word"	Comment	Example
17	diverted		
18	disruption		
19	limited operation		
20	discontinued operation		
21	irregular traffic		
22	wagon order changed		
23	train shortened		
24	additional ride		
25	replacement ride		
26	temporarily non stopping		
27	temporary stopplace		
255	undefined status		

9.12 pts044: StopPlaceUsage

Table 35 enumerates the possible values for type pts044: StopPlaceUsage.

Table 35 — pts044: StopPlaceUsage

Code	Reference-English "word"	Comment	Example
0	unknown		
1	origin		
2	destination		
3	intermediate		
4	leg board		
5	leg intermediate		
6	leg alight		
7	first route point		
8	last route point		
9	affected stopplace		
10	presented stopplace		
255	undefined stopplace usage		

9.13 pts045: Occupancy

Table 36 enumerates the possible values for type pts045: Occupancy.

Table 36 — pts045: Occupancy

Code	Reference-English "word"	Comment	Example
0	unknown		
1	many seats available		
2	few seats available		
3	no seats available		
4	standing available		
5	full		
255	undefined occupancy		

9.14 pts100:SubmodeOfTransport

The SubmodeOfTransport defines the generic type applying different tables according to the different ModeOfTransport.

Valid entries for attributes of this type are listed in the tables:

Table 37 — pts101: AirService

Table 38 — pts102: GondolaCableCarService

Table 39 — pts105: RailwayService

Table 40 — pts106: UrbanRailwayService

Table 41 — pts110: BusService

Table 42 — pts112: CoachService

Table 43 — pts115: WaterTransportService

9.15 pts101: AirService

Table 37 enumerates the possible values for type pts101: AirService.

Table 37 — pts101: AirService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	international air service		
2	national air service		
3	intercontinental air service		
4	national scheduled air service		
5	shuttle air service		
6	intercontinental air charter service		
7	international air charter		
8	round trip air charter service		

Table 37 *(continued)*

Code	Reference-English "word"	Comment	Example
9	sightseeing air service		
10	helicopter air service		
11	domestic air charter flight		
12	Schengen area air service		
13	airship service		
14	on demand service		
255	undefined air service		

9.16 pts102: GondolaCableCarService

Table 38 enumerates the possible values for type pts102: GondolaCableCarService.

Table 38 — pts102: GondolaCableCarService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	scheduled		
2	unscheduled		
255	undefined gondola cable car service		

9.17 pts105: RailwayService

Table 39 enumerates the possible values for type pts105: RailwayService.

Table 39 — pts105: RailwayService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	high speed rail service		
2	long distance international rail service		
3	long distance rail service		
4	inter regional express rail service		
5	inter regional rail service		
6	sleeper rail service		
7	regional express rail service		
8	regional rail service		
9	tourist railway service		
10	rail shuttle service		
11	suburban rail service		
12	suburban night rail service		
13	replacement rail service		
14	special rail service		

Table 39 *(continued)*

Code	Reference-English "word"	Comment	Example
15	lorry transport rail service		
16	vehicle transport rail service		
17	vehicle tunnel transport rail service		
18	additional rail service		
255	undefined rail service		

9.18 pts106: UrbanRailwayService

Table 40 enumerates the possible values for type pts106: UrbanRailwayService.

Table 40 — pts106: UrbanRailwayService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	metro service		
2	night metro service		
3	express metro service		
4	tram service		
5	city tram service		
6	regional tram service		
7	sightseeing tram service		
8	night tram service		
9	shuttle tram service		
255	undefined urban railway service		

9.19 pts110: BusService

Table 41 enumerates the possible values for type pts110: BusService.

Table 41 — pts110: BusService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	regional bus service		
2	additional bus service		
3	express bus service		
4	stopping bus service		
5	local bus service		
6	night bus service		
7	post bus service		
8	special needs bus service		
9	mobility bus service		

Table 41 *(continued)*

Code	Reference-English "word"	Comment	Example
10	mobility bus for registered disabled service		
11	sightseeing bus service		
12	shuttle bus service		
13	school bus service		
14	school and public service bus service		
15	rail replacement bus service		
16	demand and response bus service		
255	undefined bus service		

9.20 pts112: CoachService

Table 42 enumerates the possible values for type pts112: CoachService.

Table 42 — pts112: CoachService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	international coach service		
2	national coach service		
3	shuttle coach service		
4	regional coach service		
5	additional coach service		
6	night coach service		
7	special coach service		
8	sightseeing coach service		
9	tourist coach service		
10	commuter coach service		
11	on demand service		
255	undefined coach service		

9.21 pts115: WaterTransportService

Table 43 enumerates the possible values for type pts115: WaterTransportService.

Table 43 — pts115: WaterTransportService

Code	Reference-English "word"	Comment	Example
0	unknown		
1	international car ferry service		
2	national car ferry service		
3	regional car ferry service		
4	local car ferry service		

Table 43 *(continued)*

Code	Reference-English "word"	Comment	Example
5	international passenger ferry service		
6	national passenger ferry service		
7	regional passenger ferry service		
8	local passenger ferry service		
9	post boat service		
10	train ferry service		
11	road link ferry service		
12	airport link ferry service		
13	car high speed ferry service		
14	passenger high speed ferry service		
15	scheduled boat service		
16	scheduled express boat service		
17	additional boat service		
18	sightseeing boat service		
19	school boat service		
20	river bus service		
21	scheduled ferry service		
22	shuttle ferry service		
255	undefined water transport service		

Annex A
(normative)

TPEG application, TPEG-Binary Representation

A.1 Message components

A.1.1 List of generic component IDs

Table A.1 lists the generic component IDs.

Table A.1 — List of generic component IDs

Name	ID
PTSMessage	0
MessageManagementLink	1
MasterMessageLink	2
MessagePartLink	3
GeoLocation	4
AlertRequest	5
StopEventRequest	6
TripInfoRequest	7
Alert	8
StopEventForLine	9
StopEventForPlace	10
TripInfoResponse	11
DiversionLink	12

A.1.2 PTSMessage

Table A.2 shows the binary representation of the PTSMessage structure.

Table A.2 — Binary representation of PTSMessage structure

<PTSMessage(0)>:=	
<IntUnTi>(0),	ID of this component
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes
ordered {	
<MMCSwitch>(mmc),	
n ***<PtRequest>**(ptRequest)[0..1],	Component to transmit a request for public transport information.
n ***<Alert>**(alert)[0..1],	Component to transmit alert information.
n ***<StopEvent>**(stopEvent)[0..1],	Component to transmit information for a departure/arrival event (timetable) at a stopPlace or for a Line.

Table A.2 *(continued)*

n ***<TripInfoResponse>**(tripInfoResponse)[0..1],	Component to transmit the response to a TripInfoRequest.
n ***<GeoLocation>**(location)[0..1]	Map-related location referencing information for this PTS message.
};	

A.1.3 MMCSwitch

Table A.3 shows the binary representation of the MMCSwitch structure.

Table A.3 — Binary representation of MMCSwitch structure

<MMCSwitch(x)>:=	
<IntUnTi>(x),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr);	Number of bytes in attributes.

A.1.4 MessageManagementLink

Table A.4 shows the binary representation of the MessageManagementLink structure.

Table A.4 — Binary representation of MessageManagementLink structure

<MessageManagementLink(1)<MMCSwitch()>>:=	
External**<MessageManagementContainer(1)>**;	See MessageManagementContainer specification,

A.1.5 MasterMessageLink

Table A.5 shows the binary representation of the MasterMessageLink structure.

Table A.5 — Binary representation of MasterMessageLink structure

<MasterMessageLink(2)<MMCSwitch()>>:=	
External**<MMCMasterMessage(2)>**;	See MMCMasterMessage specification

A.1.6 MessagePartLink

Table A.6 shows the binary representation of the MessagePartLink structure.

Table A.6 — Binary representation of MessagePartLink structure

<MessagePartLink(3)<MMCSwitch()>>:=	
External**<MMCMessagePart(3)>**;	See MMCMessagePart specification

A.1.7 GeoLocation

Table A.7 shows the binary representation of the GeoLocation structure.

Table A.7 — Binary representation of GeoLocation structure

<GeoLocation(4)>:=	
External**<LocationReferencingContainer(4)>**;	See LocationReferencingContainer specification

A.1.8 PtRequest

Table A.8 shows the binary representation of the PtRequest structure.

Table A.8 — Binary representation of PtRequest structure

<PtRequest(x)>:=	
<IntUnTi>(x),	ID of this component
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator
<IntUnLoMB>(lengthAttr);	Number of bytes in attributes

A.1.9 AlertRequest

Table A.9 shows the binary representation of the AlertRequest structure.

Table A.9 — Binary representation of AlertRequest structure

<AlertRequest(5)<PtRequest(5)>>:=	
<IntUnTi>(5),	ID of this component
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(allAlerts),	True if an AlertRequest is issued for all available alerts; false otherwise. If false, the remaining attributes specify the AlertRequest.
if (bit 1 of selector is set)	
<LocalisedShortString>(stopPlaceName),	Well-known name of the StopPlace for which an AlertRequest is issued.
if (bit 2 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<pts001:ModeOfTransport>**(ptMode),	List of the public transportation modes for which an AlertRequest is issued.
}	
if (bit 3 of selector is set)	
<LineIdentity>(lineIdentity);	Description of the Line for which an AlertRequest is issued.

A.1.10 StopEventRequest

Table A.10 shows the binary representation of the StopEventRequest structure.

Table A.10 — Binary representation of StopEventRequest structure

<StopEventRequest(6)<PtRequest(6)>>:=	
<IntUnTi>(6),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes.
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(departure),	True if a StopEventRequest is issued for departures; false if a StopEventRequest is issued for arrivals.
if (bit 1 of selector is set)	
<LocalisedShortString>(stopPlaceName),	In case of a StopEventRequest for a StopPlace, describes the well-known name of the StopPlace.

Table A.10 *(continued)*

if (bit 2 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<pts001:ModeOfTransport>**(ptMode),	In case of a StopEventRequest for a StopPlace, lists the modes of transport to be considered for the request.
}	
if (bit 3 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<LineIdentity>**(lineIdentity),	In case of a StopEventRequest for a Line, description of the Line to be considered for the request.
}	
if (bit 4 of selector is set)	
<DateTime>(startTime),	Start of the timeframe for which the request is issued.
if (bit 5 of selector is set)	
<DateTime>(endTime);	Stop of the timeframe for which the request is issued.

A.1.11 TripInfoRequest

Table A.11 shows the binary representation of the TripInfoRequest structure.

Table A.11 — Binary representation of TripInfoRequest structure

<TripInfoRequest(7)<PtRequest(7)>>:=	
<IntUnTi>(7),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes.
<Route>(tripDetails),	Description of the whole journey from passenger origin to passenger destination in one or more TripLegs (describing origin and destination of a trip and potentially one or more vias).
<DateTime>(time),	Contains either Departure time at origin or Arrival time at destination.
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(departure),	True to indicate that time attribute contains Departure time at origin, false to indicate that time attribute contains Arrival time at destination.
if (bit 1 of selector is set)	
<IntUnTi>(numberOfResults),	Attribute to control the number of trip results before/after a point in time. May not be used when both departure time at origin and arrival time at destination are set.
if (bit 2 of selector is set)	
<TripPreferences>(tripPreferences);	User preferences to be considered in trip calculation.

A.1.12 Alert

Table A.12 shows the binary representation of the Alert structure.

Table A.12 — Binary representation of Alert structure

<Alert(8)>:=	
<IntUnTi>(8),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes.
<IntUnLoMB>(n),	
n ***<AlertFor>**(alertFor),	Describes the StopPlace, Line, Route, Area, etc. for which the alert is issued.
<pts037:AlertEvent>(alertEvent),	Information on the actual event (e.g. closure of a line, long delays of a line, cancellation of a stop).
BitArray(selector),	
if (bit 0 of selector is set)	
<pts038:AlertCause>(alertCause),	Information on the cause of the event.
if (bit 1 of selector is set)	
<Duration>(delay),	In case of a delay, this component can be used to provide information about its duration (in minutes).
if (bit 2 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<LocalisedLongString>**(alertText),	Free text option to provide additional description of the alert.
}	
if (bit 3 of selector is set)	
<pts039:AdviceType>(diversionAdvice),	Information on a potential diversion, e.g. to use an alternative route.
ordered {	
n ***<DiversionLink>**(diversionLink)[0..1]	Map-related location referencing information for a potential diversion.
};	

A.1.13 DiversionLink

Table A.13 shows the binary representation of the DiversionLink structure.

Table A.13 — Binary representation of DiversionLink structure

<DiversionLink(12)>:=	
External**<LocationReferencingContainer(12)>**;	See LocationReferencingContainer specification.

A.1.14 StopEvent

Table A.14 shows the binary representation of the StopEvent structure.

Table A.14 — Binary representation of StopEvent structure

<StopEvent(x)>:=	
<IntUnTi>(x),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr);	Number of bytes in attributes.

A.1.15 StopEventForPlace

Table A.15 shows the binary representation of the StopEventForPlace structure.

Table A.15 — Binary representation of StopEventForPlace structure

<StopEventForPlace(10)<StopEvent(10)>>:=	
<IntUnTi>(10),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes.
<IntUnLoMB>(n),	
n ***<CallAtStopForPlace>**(callAtStops),	Provides information on departure/arrival events for each relevant Line (a stop timetable).
BitArray(selector),	
if (bit 0 of selector is set)	
<StopPlace>(stopPlace);	Describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.

A.1.16 StopEventForLine

Table A.16 shows the binary representation of the StopEventForLine structure.

Table A.16 — Binary representation of StopEventForLine structure

<StopEventForLine(9)<StopEvent(9)>>:=	
<IntUnTi>(9),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes.
<IntUnLoMB>(n),	
n ***<CallAtStopForLine>**(callAtStops),	Provides information on departure/arrival events for each relevant StopPlace (along the Line).
BitArray(selector),	
if (bit 0 of selector is set)	
<LineIdentity>(line);	Description of a Line.

A.1.17 TripInfoResponse

Table A.17 shows the binary representation of the TripInfoResponse structure.

Table A.17 — Binary representation of TripInfoResponse structure

<TripInfoResponse(11)>:=	
<IntUnTi>(11),	ID of this component.
<IntUnLoMB>(lengthComp),	Number of bytes in component, excluding the id and lengthComp indicator.
<IntUnLoMB>(lengthAttr),	Number of bytes in attributes.
<IntUnLoMB>(n),	
n ***<TripLegStructure>**(tripLegs),	Description of one or more TripLegs for the trip. A TripLeg being a single stage of a Trip that is made without change of ModeOfTransport or Line (i.e. between each Transfer).
BitArray(selector),	
if (bit 0 of selector is set)	
{	

Table A.17 *(continued)*

<IntUnLoMB>(n),	
n ***<LocalisedShortString>**(tripDescription),	Descriptive text for a Trip.
}	
if (bit 1 of selector is set)	
<TimeInfo>(timeInformationForTrip);	Description of scheduled and/or actual departure and/or arrival times for a Trip.

A.2 PTS Datatypes

A.2.1 AlertFor

Table A.18 shows the binary representation of the AlertFor structure.

Table A.18 — Binary representation of AlertFor structure

<AlertFor>:=	
<pts036:AlertForType>(alertForType),	Information on the type of entity for which the alert is issued.
BitArray(selector),	
if (bit 0 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<StopPlace>**(stopPlace),	Describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.
}	
if (bit 1 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<LineIdentity>**(lineIdentity),	Description of a Line.
}	
if (bit 2 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<Route>**(route),	An ordered list of located StopPlaces defining one single path through the road or rail network.
}	
if (bit 3 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<PtServiceDescription>**(ptServiceDescription),	Information on public transport service.
}	
if (bit 4 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<LocalisedShortString>**(additionalDescription)	Textual description of an area.
};	

A.2.2 LineIdentity

Table A.19 shows the binary representation of the LineIdentity structure.

Table A.19 — Binary representation of LineIdentity structure

<LineIdentity>:=	
<ShortString>(publishedLineName),	Line name or service description as known to the public.
<IntUnLoMB>(n),	$\{0 \leq n \leq 2\}$
n ***<StopPlace>**(lineDirection),	StartPoint and Endpoint of a Line.
if (bit 1 of selector is set)	
<ShortString>(lineId);	Identifier of a Line.

A.2.3 CallAtStopForLine

Table A.20 shows the binary representation of the CallAtStopForLine structure.

Table A.20 — Binary representation of CallAtStopForLine structure

<CallAtStopForLine>:=	
<StopPlace>(stopPlace),	Describes a place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names.
<CallAtStopInfo>(callAtStopInfo);	Provides information on departure/arrival events for one certain StopPlace.

A.2.4 CallAtStopForPlace

Table A.21 shows the binary representation of the CallAtStopForPlace structure.

Table A.21 — Binary representation of CallAtStopForPlace structure

<CallAtStopForPlace>:=	
<LineIdentity>(lineIdentity),	Description of a Line.
<CallAtStopInfo>(callAtStopInfo);	Provides information on departure/arrival events for one certain Line.

A.2.5 CallAtStopInfo

Table A.22 shows the binary representation of the CallAtStopInfo structure.

Table A.22 — Binary representation of CallAtStopInfo structure

<CallAtStopInfo>:=	
BitArray(selector),	
if (bit 0 of selector is set)	
<pts043:ServiceStatus>(serviceStatus),	Describes the status of the service.
if (bit 1 of selector is set)	
<TimeInfo>(timeInformation),	Description of scheduled or estimated departure times or both and the same applies to arrival times.
if (bit 2 of selector is set)	
<StopPoint>(stopPoint),	Describes a physical area within a StopPlace (e.g. a platform)
if (bit 3 of selector is set)	
{	

Table A.22 *(continued)*

<IntUnLoMB>(n),	
n ***<LocalisedShortString>**(commentCallAtStop)	Additional descriptive text.
};	

A.2.6 OperatorContact

Table A.23 shows the binary representation of the OperatorContact structure.

Table A.23 — Binary representation of OperatorContact structure

<OperatorContact>:=	
BitArray(selector),	
if (bit 0 of selector is set)	
<pts030:ContactType>(operatorContactType),	Information on the type of possibility to contact the operator.
if (bit 1 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<LocalisedShortString>**(operatorContactInfo)	Description of the contact information for the operator.
};	

A.2.7 PtServiceDescription

Table A.24 shows the binary representation of the PtServiceDescription structure.

Table A.24 — Binary representation of PtServiceDescription structure

<PtServiceDescription>:=	
<pts001:ModeOfTransport>(ptMode),	Defines the public transport mode.
BitArray(selector),	
if (bit 0 of selector is set)	
<pts100:SubmodeOfTransport>(ptSubmode),	Defines the public transport submode.
if (bit 1 of selector is set)	
<ShortString>(operator),	Operator name as known to the public.
if (bit 2 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<OperatorContact>**(operatorContact),	Description of a contact for the operator.
}	
if (bit 3 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<LocalisedShortString>**(serviceAdditionalInformation)	Additional descriptive information about the public transport service.
};	

A.2.8 Route

Table A.25 shows the binary representation of the Route structure.

Table A.25 — Binary representation of Route structure

<Route>:=	
<IntUnLoMB>(n),	{2 ≤ *n* ≤ *}
n ***<StopPlace>**(route),	Array containing the StopPlaces on the Route.
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(bothDirection);	Only relevant for Alert: true when an Alert is issued for both directions of the route.

A.2.9 StopPlace

Table A.26 shows the binary representation of the StopPlace structure.

Table A.26 — Binary representation of StopPlace structure

<StopPlace>:=	
<pts044:StopPlaceUsage>(stopPlaceUsage),	Identifies the "role" a StopPlace has in a Trip (e.g. origin, destination, intermediate) or for an Alert (e.g. temporarily non-stopping) or on a Line (e.g. temporary stopplace, first route point).
<IntUnLoMB>(n),	
n ***<LocalisedShortString>**(stopPlaceName),	Well-known name of the StopPlace.
BitArray(selector),	
if (bit 0 of selector is set)	
<ShortString>(stopPlaceRef),	Reference key for the StopPlace.
if (bit 1 of selector is set)	
<pts041:StopPlaceType>(stopPlaceType),	Information on the type of StopPlace (e.g. railway station, bus station).
if (bit 2 of selector is set)	
<StopPoint>(stopPoint),	Describes a physical area within a StopPlace (e.g. a platform).
if (bit 3 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<pts040:AccessFeatureType>**(accessFeature),	Describes accessibility features associated with this StopPlace.
}	
if (bit 4 of selector is set)	
<pts042:FacilityType>(facilityType);	Named amenity available to the public at this StopPlace.

A.2.10 StopPoint

Table A.27 shows the binary representation of the StopPoint structure.

Table A.27 — Binary representation of StopPoint structure

<StopPoint>:=	
<pts017:ServiceDeliveryPointType>(stopPointType),	Information on the type of StopPoint (e.g. gate, platform).
<ShortString>(stopPointName),	Well-known name of the StopPoint.
BitArray(selector),	
if (bit 0 of selector is set)	
<ShortString>(stopPointRef);	Reference key for the StopPoint.

A.2.11 TimeInfo

Table A.28 shows the binary representation of the TimeInfo structure.

Table A.28 — Binary representation of TimeInfo structure

<TimeInfo>:=	
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(estimatedTimeKnown),	True if an expected/estimated time is known; False otherwise.
if (bit 1 of selector is set)	
<DateTime>(timetabledDepartureTime),	Departure time according to planned timetable.
if (bit 2 of selector is set)	
<DateTime>(timetabledArrivalTime),	Arrival time according to planned timetable.
if (bit 3 of selector is set)	
<Duration>(travelTime),	Travel time for a trip (in case of TripInfo); travel time for a TripLeg (in case of TripLegStructure).
if (bit 4 of selector is set)	
<DateTime>(estimatedDepartureTime),	Expected/estimated departure time at the StopPoint.
if (bit 5 of selector is set)	
<DateTime>(estimatedArrivalTime);	Expected/estimated arrival time at the StopPoint.

A.2.12 TripLegStructure

Table A.29 shows the binary representation of the TripLegStructure structure.

Table A.29 — Binary representation of TripLegStructure structure

<TripLegStructure>:=	
<LineIdentity>(lineIdentity),	Description of a Line.
<Route>(tripLeg),	A single stage of a Trip that is made without change of ModeOfTransport or Line (i.e. between each Transfer).
BitArray(selector),	
if (bit 0 of selector is set)	
<PtServiceDescription>(ptServiceDescription),	Description of a public transport service.
if (bit 1 of selector is set)	
<TimeInfo>(timeInformation),	Description of scheduled or actual departure times or both at the starting point of the TripLeg and the same applies to the arrival times at the ending point of the TripLeg, as well as the travel time.
if (bit 2 of selector is set)	
<pts045:Occupancy>(occupancy);	How full the vehicle is. If omitted, not known.

A.2.13 TripPreferences

Table A.30 shows the binary representation of the TripPreferences structure.

Table A.30 — Binary representation of TripPreferences structure

<TripPreferences>:=	
BitArray(selector),	
if (bit 0 of selector is set)	
<Boolean>(levelEntrance),	If set to true the user needs vehicles with level entrance between platform and vehicle, e.g. for wheel-chair access. Liftequipped vehicles or stationary lifts at the platform may be sufficient. Default is false.
if (bit 1 of selector is set)	
<Boolean>(bikeTransport),	If set to true the user wants to carry a bike on public transport. Default is false.
if (bit 2 of selector is set)	
<IntUnTi>(walkSpeed),	Deviation from average walk speed in percent. 100 % percent means average speed. Values less than 100 % mean slower, greater than 100 % faster. Default is 100 %.
if (bit 3 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<pts040:AccessFeatureType>**(accessFacilities),	Specific accessibility/mobility requirements set to access paths and connection paths (with respect to the physical restrictions that apply to them).
}	
if (bit 4 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<pts001:ModeOfTransport>**(excludedModes),	Public transport modes to be excluded in trip calculation.
}	
if (bit 5 of selector is set)	
{	
<IntUnLoMB>(n),	
n ***<pts100:SubmodeOfTransport>**(excludedSubModes)	Public transport submodes to be excluded in trip calculation.
};	

Annex B
(normative)

TPEG application, tpegML Representation

B.1 Message components

B.1.1 PTSMessage

```
<xs:element name="PTSMessage" type="PTSMessage"/>
<xs:complexType name="PTSMessage">
 <xs:complexContent>
 <xs:extension base="tsf:ApplicationRootMessageML">
 <xs:sequence>
 <xs:element name="mmc" type="MMCSwitch"/>
 <xs:element name="ptRequest" type="PtRequest" minOccurs="0"/>
 <xs:element name="alert" type="Alert" minOccurs="0"/>
 <xs:element name="stopEvent" type="StopEvent" minOccurs="0"/>
 <xs:element name="tripInfoResponse" type="TripInfoResponse" minOccurs="0"/>
 <xs:element name="location" type="lrc:LocationReferencingContainer" minOccurs="0"/>
 </xs:sequence>
 </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

B.1.2 MMCSwitch

```
<xs:complexType name="MMCSwitch">
 <xs:sequence>
 <xs:choice minOccurs="1" maxOccurs="1">
 <xs:element name="optionMessageManagementLink" type="mmc:MessageManagementContainer"
 minOccurs="1" maxOccurs="1"/>
 <xs:element name="optionMasterMessageLink" type="mmc:MMCMasterMessage" minOccurs="1"
 maxOccurs="1"/>
 <xs:element name="optionMessagePartLink" type="mmc:MMCMessagePart" minOccurs="1" max
Occurs="1"/>
 </xs:choice>
 </xs:sequence>
</xs:complexType>
```

B.1.3 PtRequest

```
<xs:complexType name="PtRequest">
 <xs:sequence>
 <xs:choice minOccurs="1" maxOccurs="1">
 <xs:element name="optionAlertRequest" type="AlertRequest" minOccurs="1" maxOccurs="1"/>
 <xs:element name="optionStopEventRequest" type="StopEventRequest" minOccurs="1"
maxOccurs="1"/>
 <xs:element name="optionTripInfoRequest" type="TripInfoRequest" minOccurs="1" maxOccurs="1"/>
 </xs:choice>
 </xs:sequence>
</xs:complexType>
```

B.1.4 AlertRequest

```
<xs:complexType name="AlertRequest">
 <xs:sequence>
 <xs:element name="allAlerts" type="tdt:Boolean"/>
 <xs:element name="stopPlaceName" type="tdt:LocalisedShortString" minOccurs="0"/>
 <xs:element name="ptMode" type="pts001_ModeOfTransport" minOccurs="0" maxOccurs="unbounded"/>
```

```
 <xs:element name="lineIdentity" type="LineIdentity" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.1.5 StopEventRequest

```
<xs:complexType name="StopEventRequest">
 <xs:sequence>
 <xs:element name="departure" type="tdt:Boolean"/>
 <xs:element name="stopPlaceName" type="tdt:LocalisedShortString" minOccurs="0"/>
 <xs:element name="ptMode" type="pts001_ModeOfTransport" minOccurs="0" maxOccurs="unbounded"/>
 <xs:element name="lineIdentity" type="LineIdentity" minOccurs="0" maxOccurs="unbounded"/>
 <xs:element name="startTime" type="tdt:DateTime" minOccurs="0"/>
 <xs:element name="endTime" type="tdt:DateTime" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.1.6 TripInfoRequest

```
<xs:complexType name="TripInfoRequest">
 <xs:sequence>
 <xs:element name="tripDetails" type="Route"/>
 <xs:element name="time" type="tdt:DateTime"/>
 <xs:element name="departure" type="tdt:Boolean"/>
 <xs:element name="numberOfResults" type="tdt:IntUnTi" minOccurs="0"/>
 <xs:element name="tripPreferences" type="TripPreferences" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.1.7 Alert

```
<xs:complexType name="Alert">
 <xs:sequence>
 <xs:element name="alertFor" type="AlertFor" maxOccurs="unbounded"/>
 <xs:element name="alertEvent" type="pts037_AlertEvent"/>
 <xs:element name="alertCause" type="pts038_AlertCause" minOccurs="0"/>
 <xs:element name="delay" type="tdt:Duration" minOccurs="0"/>
 <xs:element name="alertText" type="tdt:LocalisedLongString" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="diversionAdvice" type="pts039_AdviceType" minOccurs="0"/>
 <xs:element name="diversionLink" type="lrc:LocationReferencingContainer" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.1.8 StopEvent

```
<xs:complexType name="StopEvent">
 <xs:sequence>
 <xs:choice minOccurs="1" maxOccurs="1">
 <xs:element name="optionStopEventForPlace" type="StopEventForPlace" minOccurs="1"
maxOccurs="1"/>
 <xs:element name="optionStopEventForLine" type="StopEventForLine" minOccurs="1"
maxOccurs="1"/>
 </xs:choice>
 </xs:sequence>
</xs:complexType>
```

B.1.9 StopEventForPlace

```
<xs:complexType name="StopEventForPlace">
 <xs:sequence>
 <xs:element name="callAtStops" type="CallAtStopForPlace" maxOccurs="unbounded"/>
 <xs:element name="stopPlace" type="StopPlace" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.1.10 StopEventForLine

```
<xs:complexType name="StopEventForLine">
 <xs:sequence>
 <xs:element name="callAtStops" type="CallAtStopForLine" maxOccurs="unbounded"/>
 <xs:element name="line" type="LineIdentity" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.1.11 TripInfoResponse

```
<xs:complexType name="TripInfoResponse">
 <xs:sequence>
 <xs:element name="tripLegs" type="TripLegStructure" maxOccurs="unbounded"/>
 <xs:element name="tripDescription" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="timeInformationForTrip" type="TimeInfo" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.2 Datatypes

B.2.1 AlertFor

```
<xs:complexType name="AlertFor">
 <xs:sequence>
 <xs:element name="alertForType" type="pts036_AlertForType"/>
 <xs:element name="stopPlace" type="StopPlace" minOccurs="0" maxOccurs="unbounded"/>
 <xs:element name="lineIdentity" type="LineIdentity" minOccurs="0" maxOccurs="unbounded"/>
 <xs:element name="route" type="Route" minOccurs="0" maxOccurs="unbounded"/>
 <xs:element name="ptServiceDescription" type="PtServiceDescription" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="additionalDescription" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

B.2.2 LineIdentity

```
<xs:complexType name="LineIdentity">
 <xs:sequence>
 <xs:element name="publishedLineName" type="tdt:ShortString"/>
 <xs:element name="lineDirection" type="StopPlace" minOccurs="0" maxOccurs="2"/>
 <xs:element name="lineId" type="tdt:ShortString" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.2.3 CallAtStopForLine

```
<xs:complexType name="CallAtStopForLine">
 <xs:sequence>
 <xs:element name="stopPlace" type="StopPlace"/>
 <xs:element name="callAtStopInfo" type="CallAtStopInfo"/>
 </xs:sequence>
</xs:complexType>
```

B.2.4 CallAtStopForPlace

```
<xs:complexType name="CallAtStopForPlace">
 <xs:sequence>
 <xs:element name="lineIdentity" type="LineIdentity"/>
 <xs:element name="callAtStopInfo" type="CallAtStopInfo"/>
 </xs:sequence>
</xs:complexType>
```

B.2.5 CallAtStopInfo

```
<xs:complexType name="CallAtStopInfo">
 <xs:sequence>
 <xs:element name="serviceStatus" type="pts043_ServiceStatus" minOccurs="0"/>
 <xs:element name="timeInformation" type="TimeInfo" minOccurs="0"/>
 <xs:element name="stopPoint" type="StopPoint" minOccurs="0"/>
 <xs:element name="commentCallAtStop" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

B.2.6 OperatorContact

```
<xs:complexType name="OperatorContact">
 <xs:sequence>
 <xs:element name="operatorContactType" type="pts030_ContactType" minOccurs="0"/>
 <xs:element name="operatorContactInfo" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

B.2.7 PtServiceDescription

```
<xs:complexType name="PtServiceDescription">
 <xs:sequence>
 <xs:element name="ptMode" type="pts001_ModeOfTransport"/>
 <xs:element name="ptSubmode" type="pts100_SubmodeOfTransport" minOccurs="0"/>
 <xs:element name="operator" type="tdt:ShortString" minOccurs="0"/>
 <xs:element name="operatorContact" type="OperatorContact" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="serviceAdditionalInformation" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

B.2.8 Route

```
<xs:complexType name="Route">
 <xs:sequence>
 <xs:element name="route" type="StopPlace" minOccurs="2" maxOccurs="unbounded"/>
 <xs:element name="bothDirection" type="tdt:Boolean"/>
 </xs:sequence>
</xs:complexType>
```

B.2.9 StopPlace

```
<xs:complexType name="StopPlace">
 <xs:sequence>
 <xs:element name="stopPlaceUsage" type="pts044_StopPlaceUsage"/>
 <xs:element name="stopPlaceName" type="tdt:LocalisedShortString" maxOccurs="unbounded"/>
 <xs:element name="stopPlaceRef" type="tdt:ShortString" minOccurs="0"/>
 <xs:element name="stopPlaceType" type="pts041_StopPlaceType" minOccurs="0"/>
 <xs:element name="stopPoint" type="StopPoint" minOccurs="0"/>
 <xs:element name="accessFeature" type="pts040_AccessFeatureType" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="facilityType" type="pts042_FacilityType" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.2.10 StopPoint

```
<xs:complexType name="StopPoint">
 <xs:sequence>
 <xs:element name="stopPointType" type="pts017_ServiceDeliveryPointType"/>
 <xs:element name="stopPointName" type="tdt:ShortString"/>
 <xs:element name="stopPointRef" type="tdt:ShortString" minOccurs="0"/>
 </xs:sequence>
```

```
</xs:complexType>
```

B.2.11 TimeInfo

```
<xs:complexType name="TimeInfo">
 <xs:sequence>
 <xs:element name="estimatedTimeKnown" type="tdt:Boolean"/>
 <xs:element name="timetabledDepartureTime" type="tdt:DateTime" minOccurs="0"/>
 <xs:element name="timetabledArrivalTime" type="tdt:DateTime" minOccurs="0"/>
 <xs:element name="travelTime" type="tdt:Duration" minOccurs="0"/>
 <xs:element name="estimatedDepartureTime" type="tdt:DateTime" minOccurs="0"/>
 <xs:element name="estimatedArrivalTime" type="tdt:DateTime" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.2.12 TripLegStructure

```
<xs:complexType name="TripLegStructure">
 <xs:sequence>
 <xs:element name="lineIdentity" type="LineIdentity"/>
 <xs:element name="tripLeg" type="Route"/>
 <xs:element name="ptServiceDescription" type="PtServiceDescription" minOccurs="0"/>
 <xs:element name="timeInformation" type="TimeInfo" minOccurs="0"/>
 <xs:element name="occupancy" type="pts045_Occupancy" minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

B.2.13 TripPreferences

```
<xs:complexType name="TripPreferences">
 <xs:sequence>
 <xs:element name="levelEntrance" type="tdt:Boolean"/>
 <xs:element name="bikeTransport" type="tdt:Boolean"/>
 <xs:element name="walkSpeed" type="tdt:IntUnTi" minOccurs="0"/>
 <xs:element name="accessFacilities" type="pts040_AccessFeatureType" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="excludedModes" type="pts001_ModeOfTransport" minOccurs="0"
maxOccurs="unbounded"/>
 <xs:element name="excludedSubModes" type="pts100_SubmodeOfTransport" minOccurs="0"
maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

B.3 Full PTS schema definition

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- This XML schema is generated with tpegUMLconverter 3.1, at 2019-12-19 16:09, on basis of
release 1.0 -->
<xs:schema xmlns="https://www.tisa.org/TPEG/PTS_1_0"
   targetNamespace="https://www.tisa.org/TPEG/PTS_1_0"
   xmlns:xs="https://www.w3.org/2001/XMLSchema"
   xmlns:tsf="https://www.tisa.org/TPEG/SFW_1_1"
   xmlns:tdt="https://www.tisa.org/TPEG/TPEGDataTypes_2_0"
   xmlns:mmc="https://www.tisa.org/TPEG/MMC_1_1"
   xmlns:lrc="https://www.tisa.org/TPEG/LRC_2_1"
   elementFormDefault="qualified"
   attributeFormDefault="qualified">
   <xs:import namespace="https://www.tisa.org/TPEG/SFW_1_1" schemaLocation="SFW_1_1.xsd"/>
   <xs:import namespace="https://www.tisa.org/TPEG/TPEGDataTypes_2_0" schemaLocation="TDT_2_0.
xsd"/>
   <xs:import namespace="https://www.tisa.org/TPEG/MMC_1_1" schemaLocation="MMC_1_1.xsd"/>
   <xs:import namespace="https://www.tisa.org/TPEG/LRC_2_1" schemaLocation="LRC_2_1.xsd"/>
   <xs:element name="PTSMessage" type="PTSMessage"/>
   <xs:complexType name="PTSMessage">
      <xs:complexContent>
         <xs:extension base="tsf:ApplicationRootMessageML">
            <xs:sequence>
               <xs:element name="mmc" type="MMCSwitch"/>
```

```
               <xs:element name="ptRequest" type="PtRequest" minOccurs="0"/>
               <xs:element name="alert" type="Alert" minOccurs="0"/>
               <xs:element name="stopEvent" type="StopEvent" minOccurs="0"/>
               <xs:element name="tripInfoResponse" type="TripInfoResponse" minOccurs="0"/>
               <xs:element name="location" type="lrc:LocationReferencingContainer"
minOccurs="0"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
   <xs:complexType name="MMCSwitch">
      <xs:sequence>
         <xs:choice minOccurs="1" maxOccurs="1">
            <xs:element name="optionMessageManagementLink" type="mmc:MessageManagementContai
ner" minOccurs="1" maxOccurs="1"/>
            <xs:element name="optionMasterMessageLink" type="mmc:MMCMasterMessage"
minOccurs="1" maxOccurs="1"/>
            <xs:element name="optionMessagePartLink" type="mmc:MMCMessagePart" minOccurs="1"
maxOccurs="1"/>
         </xs:choice>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="PtRequest">
      <xs:sequence>
         <xs:choice minOccurs="1" maxOccurs="1">
            <xs:element name="optionAlertRequest" type="AlertRequest" minOccurs="1"
maxOccurs="1"/>
            <xs:element name="optionStopEventRequest" type="StopEventRequest" minOccurs="1"
maxOccurs="1"/>
            <xs:element name="optionTripInfoRequest" type="TripInfoRequest" minOccurs="1"
maxOccurs="1"/>
         </xs:choice>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="AlertRequest">
      <xs:sequence>
         <xs:element name="allAlerts" type="tdt:Boolean"/>
         <xs:element name="stopPlaceName" type="tdt:LocalisedShortString" minOccurs="0"/>
         <xs:element name="ptMode" type="pts001_ModeOfTransport" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="lineIdentity" type="LineIdentity" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="StopEventRequest">
      <xs:sequence>
         <xs:element name="departure" type="tdt:Boolean"/>
         <xs:element name="stopPlaceName" type="tdt:LocalisedShortString" minOccurs="0"/>
         <xs:element name="ptMode" type="pts001_ModeOfTransport" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="lineIdentity" type="LineIdentity" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="startTime" type="tdt:DateTime" minOccurs="0"/>
         <xs:element name="endTime" type="tdt:DateTime" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="TripInfoRequest">
      <xs:sequence>
         <xs:element name="tripDetails" type="Route"/>
         <xs:element name="time" type="tdt:DateTime"/>
         <xs:element name="departure" type="tdt:Boolean"/>
         <xs:element name="numberOfResults" type="tdt:IntUnTi" minOccurs="0"/>
         <xs:element name="tripPreferences" type="TripPreferences" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="Alert">
      <xs:sequence>
         <xs:element name="alertFor" type="AlertFor" maxOccurs="unbounded"/>
         <xs:element name="alertEvent" type="pts037_AlertEvent"/>
         <xs:element name="alertCause" type="pts038_AlertCause" minOccurs="0"/>
         <xs:element name="delay" type="tdt:Duration" minOccurs="0"/>
         <xs:element name="alertText" type="tdt:LocalisedLongString" minOccurs="0"
maxOccurs="unbounded"/>
```

```
         <xs:element name="diversionAdvice" type="pts039_AdviceType" minOccurs="0"/>
         <xs:element name="diversionLink" type="lrc:LocationReferencingContainer"
minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="StopEvent">
      <xs:sequence>
         <xs:choice minOccurs="1" maxOccurs="1">
            <xs:element name="optionStopEventForPlace" type="StopEventForPlace" minOccurs="1"
maxOccurs="1"/>
            <xs:element name="optionStopEventForLine" type="StopEventForLine" minOccurs="1"
maxOccurs="1"/>
         </xs:choice>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="StopEventForPlace">
      <xs:sequence>
         <xs:element name="callAtStops" type="CallAtStopForPlace" maxOccurs="unbounded"/>
         <xs:element name="stopPlace" type="StopPlace" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="StopEventForLine">
      <xs:sequence>
         <xs:element name="callAtStops" type="CallAtStopForLine" maxOccurs="unbounded"/>
         <xs:element name="line" type="LineIdentity" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="TripInfoResponse">
      <xs:sequence>
         <xs:element name="tripLegs" type="TripLegStructure" maxOccurs="unbounded"/>
         <xs:element name="tripDescription" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="timeInformationForTrip" type="TimeInfo" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="AlertFor">
      <xs:sequence>
         <xs:element name="alertForType" type="pts036_AlertForType"/>
         <xs:element name="stopPlace" type="StopPlace" minOccurs="0" maxOccurs="unbounded"/>
         <xs:element name="lineIdentity" type="LineIdentity" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="route" type="Route" minOccurs="0" maxOccurs="unbounded"/>
         <xs:element name="ptServiceDescription" type="PtServiceDescription" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="additionalDescription" type="tdt:LocalisedShortString"
minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="LineIdentity">
      <xs:sequence>
         <xs:element name="publishedLineName" type="tdt:ShortString"/>
         <xs:element name="lineDirection" type="StopPlace" minOccurs="0" maxOccurs="2"/>
         <xs:element name="lineId" type="tdt:ShortString" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="CallAtStopForLine">
      <xs:sequence>
         <xs:element name="stopPlace" type="StopPlace"/>
         <xs:element name="callAtStopInfo" type="CallAtStopInfo"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="CallAtStopForPlace">
      <xs:sequence>
         <xs:element name="lineIdentity" type="LineIdentity"/>
         <xs:element name="callAtStopInfo" type="CallAtStopInfo"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="CallAtStopInfo">
      <xs:sequence>
         <xs:element name="serviceStatus" type="pts043_ServiceStatus" minOccurs="0"/>
         <xs:element name="timeInformation" type="TimeInfo" minOccurs="0"/>
         <xs:element name="stopPoint" type="StopPoint" minOccurs="0"/>
```

```
         <xs:element name="commentCallAtStop" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="OperatorContact">
      <xs:sequence>
         <xs:element name="operatorContactType" type="pts030_ContactType" minOccurs="0"/>
         <xs:element name="operatorContactInfo" type="tdt:LocalisedShortString" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="PtServiceDescription">
      <xs:sequence>
         <xs:element name="ptMode" type="pts001_ModeOfTransport"/>
         <xs:element name="ptSubmode" type="pts100_SubmodeOfTransport" minOccurs="0"/>
         <xs:element name="operator" type="tdt:ShortString" minOccurs="0"/>
         <xs:element name="operatorContact" type="OperatorContact" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="serviceAdditionalInformation" type="tdt:LocalisedShortString"
minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="Route">
      <xs:sequence>
         <xs:element name="route" type="StopPlace" minOccurs="2" maxOccurs="unbounded"/>
         <xs:element name="bothDirection" type="tdt:Boolean"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="StopPlace">
      <xs:sequence>
         <xs:element name="stopPlaceUsage" type="pts044_StopPlaceUsage"/>
         <xs:element name="stopPlaceName" type="tdt:LocalisedShortString"
maxOccurs="unbounded"/>
         <xs:element name="stopPlaceRef" type="tdt:ShortString" minOccurs="0"/>
         <xs:element name="stopPlaceType" type="pts041_StopPlaceType" minOccurs="0"/>
         <xs:element name="stopPoint" type="StopPoint" minOccurs="0"/>
         <xs:element name="accessFeature" type="pts040_AccessFeatureType" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="facilityType" type="pts042_FacilityType" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="StopPoint">
      <xs:sequence>
         <xs:element name="stopPointType" type="pts017_ServiceDeliveryPointType"/>
         <xs:element name="stopPointName" type="tdt:ShortString"/>
         <xs:element name="stopPointRef" type="tdt:ShortString" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="TimeInfo">
      <xs:sequence>
         <xs:element name="estimatedTimeKnown" type="tdt:Boolean"/>
         <xs:element name="timetabledDepartureTime" type="tdt:DateTime" minOccurs="0"/>
         <xs:element name="timetabledArrivalTime" type="tdt:DateTime" minOccurs="0"/>
         <xs:element name="travelTime" type="tdt:Duration" minOccurs="0"/>
         <xs:element name="estimatedDepartureTime" type="tdt:DateTime" minOccurs="0"/>
         <xs:element name="estimatedArrivalTime" type="tdt:DateTime" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="TripLegStructure">
      <xs:sequence>
         <xs:element name="lineIdentity" type="LineIdentity"/>
         <xs:element name="tripLeg" type="Route"/>
         <xs:element name="ptServiceDescription" type="PtServiceDescription" minOccurs="0"/>
         <xs:element name="timeInformation" type="TimeInfo" minOccurs="0"/>
         <xs:element name="occupancy" type="pts045_Occupancy" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="TripPreferences">
      <xs:sequence>
         <xs:element name="levelEntrance" type="tdt:Boolean"/>
         <xs:element name="bikeTransport" type="tdt:Boolean"/>
         <xs:element name="walkSpeed" type="tdt:IntUnTi" minOccurs="0"/>
```

```
         <xs:element name="accessFacilities" type="pts040_AccessFeatureType" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="excludedModes" type="pts001_ModeOfTransport" minOccurs="0"
maxOccurs="unbounded"/>
         <xs:element name="excludedSubModes" type="pts100_SubmodeOfTransport" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
   </xs:complexType>
   <xs:complexType name="pts001_ModeOfTransport">
      <xs:attribute name="table" type="xs:string" fixed="pts001_ModeOfTransport"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts017_ServiceDeliveryPointType">
      <xs:attribute name="table" type="xs:string" fixed="pts017_ServiceDeliveryPointType"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts030_ContactType">
      <xs:attribute name="table" type="xs:string" fixed="pts030_ContactType" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts036_AlertForType">
      <xs:attribute name="table" type="xs:string" fixed="pts036_AlertForType" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts037_AlertEvent">
      <xs:attribute name="table" type="xs:string" fixed="pts037_AlertEvent" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts038_AlertCause">
      <xs:attribute name="table" type="xs:string" fixed="pts038_AlertCause" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
```

```
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts039_AdviceType">
      <xs:attribute name="table" type="xs:string" fixed="pts039_AdviceType" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts040_AccessFeatureType">
      <xs:attribute name="table" type="xs:string" fixed="pts040_AccessFeatureType"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts041_StopPlaceType">
      <xs:attribute name="table" type="xs:string" fixed="pts041_StopPlaceType"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts042_FacilityType">
      <xs:attribute name="table" type="xs:string" fixed="pts042_FacilityType" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts043_ServiceStatus">
      <xs:attribute name="table" type="xs:string" fixed="pts043_ServiceStatus"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts044_StopPlaceUsage">
      <xs:attribute name="table" type="xs:string" fixed="pts044_StopPlaceUsage"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
```

```
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts045_Occupancy">
      <xs:attribute name="table" type="xs:string" fixed="pts045_Occupancy" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts100_SubmodeOfTransport">
      <xs:attribute name="table" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:string">
               <xs:enumeration value="pts101_AirService"/>
               <xs:enumeration value="pts102_GondolaCableCarService"/>
               <xs:enumeration value="pts105_RailwayService"/>
               <xs:enumeration value="pts106_UrbanRailwayService"/>
               <xs:enumeration value="pts110_BusService"/>
               <xs:enumeration value="pts112_CoachService"/>
               <xs:enumeration value="pts115_WaterTransportService"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="code" type="xs:unsignedByte" use="required"/>
   </xs:complexType>
   <xs:complexType name="pts101_AirService">
      <xs:attribute name="table" type="xs:string" fixed="pts101_AirService" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts102_GondolaCableCarService">
      <xs:attribute name="table" type="xs:string" fixed="pts102_GondolaCableCarService"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts105_RailwayService">
      <xs:attribute name="table" type="xs:string" fixed="pts105_RailwayService"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts106_UrbanRailwayService">
      <xs:attribute name="table" type="xs:string" fixed="pts106_UrbanRailwayService"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
```

```
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts110_BusService">
      <xs:attribute name="table" type="xs:string" fixed="pts110_BusService" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts112_CoachService">
      <xs:attribute name="table" type="xs:string" fixed="pts112_CoachService" use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
   <xs:complexType name="pts115_WaterTransportService">
      <xs:attribute name="table" type="xs:string" fixed="pts115_WaterTransportService"
use="required"/>
      <xs:attribute name="code" use="required">
         <xs:simpleType>
            <xs:restriction base="xs:unsignedByte">
               <xs:minInclusive value="0"/>
               <xs:maxInclusive value="255"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:attribute>
   </xs:complexType>
</xs:schema>
```

Annex C
(informative)

PTS usage examples

C.1 Introduction

The examples in this annex illustrate the usage of PTS components and attributes without claiming to be exhaustive.

The PTS specification is designed to deliver the information for three main use cases.

— Alert information.

— Timetable information, both scheduled and real time.

— Individual requests for trip information.

C.2 Alert

C.2.1 General

Each current alert is delivered by a separate message. It is allowed to deliver an Alert and a StopEvent component in a same message (see Clause C.3).

The location container of a message containing an alert delivers information about the location for which the alert is valid.

C.2.2 Alert for an underground station closure

The component Alert (see Figure C.1) contains the alert event (closed), cause of the alert (fire), the additional descriptive text and the diversion advice.

The alternative route is verbally described in the alert text and encoded as a location referencing container in the diversion link component.

The components AlertFor and PtServiceDescription describe the object and the ModeOfTransport for which the alert is relevant (underground stop place). The name (Marienplatz) and the StopPlaceRef of the affected StopPlace are given in the component StopPlace.

In this case, the attribute stopPlaceUsage has the value “affected stopplace”.

The LineIdentity components describe the relevant published line names (U3 and U6).

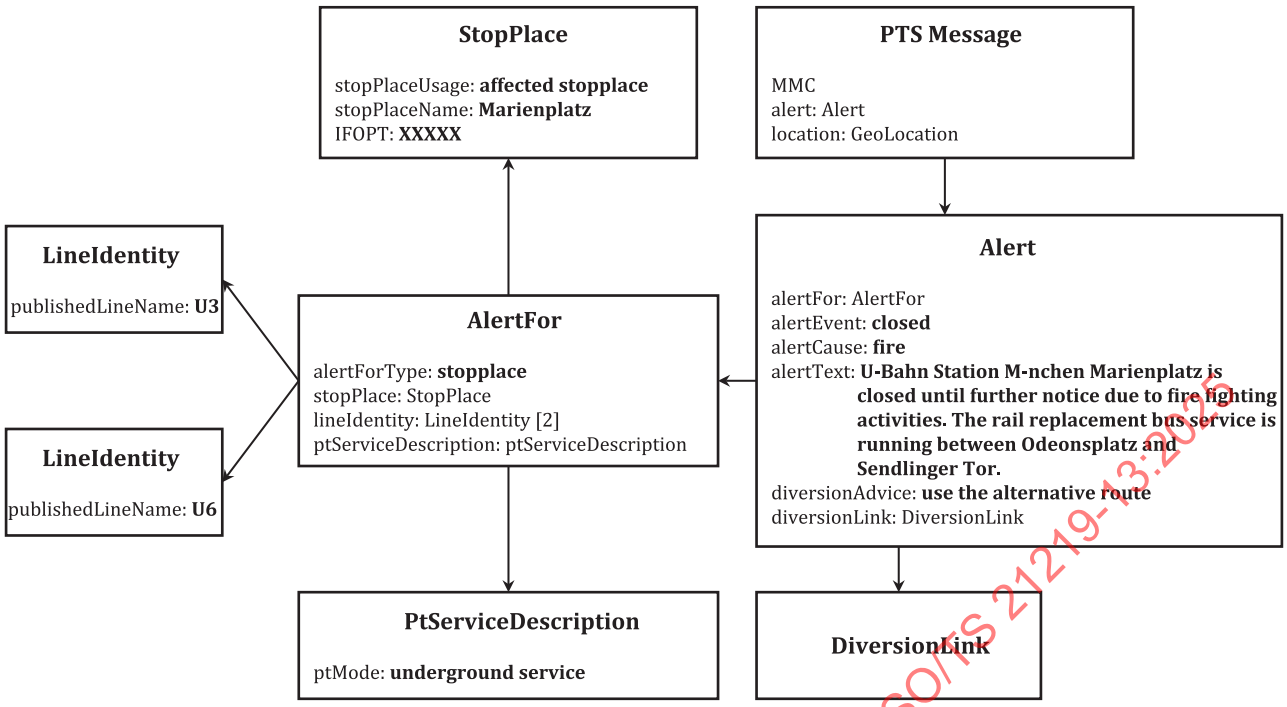

Figure C.1 — Alert for an underground station closure

C.2.3 Alert for underground delays in both directions

The component Alert (see Figure C.2) contains the alert event (delays), cause of the alert (signal failure) and the delay time (20 minutes). The component AlertFor describes the object and the public transport mode for which the alert is relevant (underground line). The component LineIdentity describes the published line name (U1).

If the attribute lineDirection is not present in LineIdentity, this implies "in both directions".

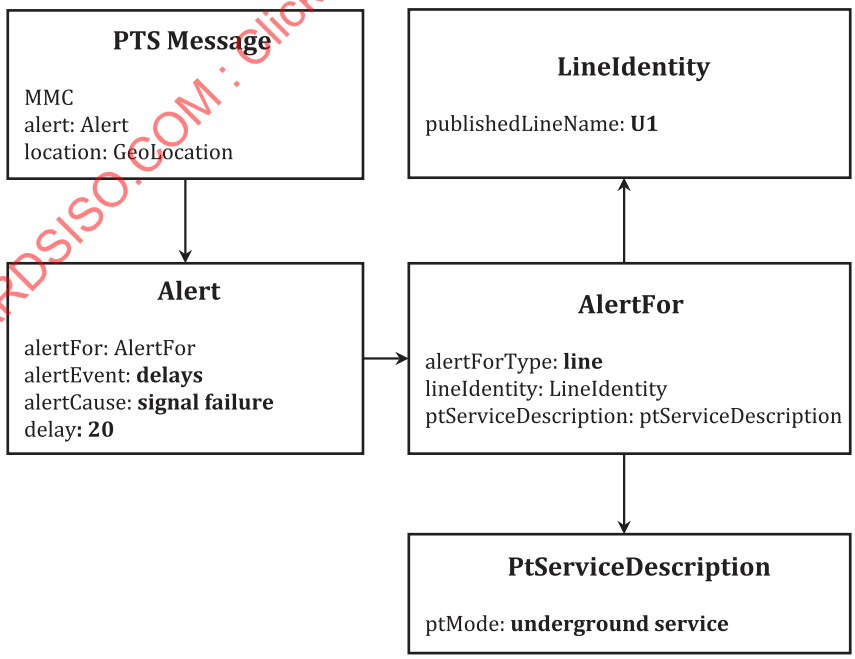

Figure C.2 — Alert for underground delays in both directions

C.2.4 Alert for underground delays in one direction

Figure C.3 uses the same structure as shown in Figure C.2 but the direction is explicitly given by origin and destination stop places (U1 from Mangfallplatz to OEZ).

The attribute stopPlaceUsage in the first StopPlace component has the value “origin” and the attribute stopPlaceUsage in the second StopPlace component has the value “destination”.

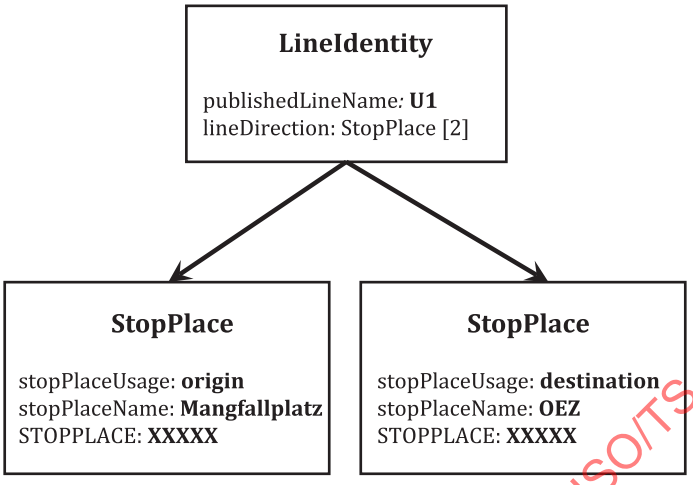

Figure C.3 — Alert for underground delays in one direction: LineIdentity component

C.2.5 Alert for a cancelled suburban railway service

The component Alert (see Figure C.4) contains the alert event (cancelled), cause of the alert (heavy snowfall) and the additional descriptive text. The components AlertFor and PtServiceDescription describe the affected public transport mode (suburban railway) and the operator (MVV). Additionally, the operator contact (telephone number) is given.

The reference to the affected area is given with the location referencing container.