DATEX II V2.1

USER GUIDE

Document version: 1.0

31 May 2012

European Commission

Directorate General for Mobility and Transport

Prepared by:			
	Date	**Comment**	**Version**
ESG5 Technical Group	31/05/2012	Updated by Loïc Blaive	1.0

Reviewed by:			
	Date	**Comment**	**Version**
ESG5 Technical Group	31/05/2012		1.0

Approved by:			
	Date	**Comment**	**Version**
ESG5 Technical Group	31/05/2012	Authorization for publication	1.0

TABLE OF CONTENTS

Introduction

1. Introduction

1.1. Objectives

This document is mainly targeted towards non-software personnel who wish to understand what DATEX II is. It represents an introduction and a help to understand other DATEX II documents (which can be downloaded from the DATEX II web site http://www.datex2.eu).

This document should also be of use to software developers as a reference document.For this latter category of readers, it is completed by a Software developer guide [**Developer guide**].

1.2. Document structure

The document is structured into 6 chapters with the following content:

- Chapter 1 “Introduction” gives a brief overview on the document objectives and structure and lists DATEX II reference documents.
- Chapter 2 “DATEX II in general” provides background information and details of important DATEX II features. It also introduces the whole set of DATEX II reference documents.
- Chapter 3 “Exchanged data” lists all possible exchanged data, introduces DATEX II terminology and presents new DATEX II concepts.
- Chapter 4 “UML data model content” presents the data model structure (packages, diagrams, classes, attributes, etc.)
- Chapter 5 “Exchange specification” explains, at a high level the exchange mechanisms and their options.
- Chapter 6: “Interchange agreement” provides help to build exchanges between a Supplier and its Clients.

Annexes provide some explanations about UML formalism and about the different location referencing systems adopted by DATEX II.

1.3. DATEX II reference documents

Reference in this document	DATEX II document	Document version	Date
[User Guide]	DATEX II v2.1 User Guide	1.0	31/05/2012
[Developer guide]	DATEX II v2.1 Software Developer Guide	1.0	31/05/2012
[Modelling methodology]	DATEX II v2.1 Modelling methodology	2.1	31/05/2012
[Data model]	DATEX II v2.1 Data model	2.1	31/05/2012
[Dictionary]	DATEX II v2.1 Data definitions	2.1	31/05/2012
[XML schema]	DATEX II v2.1 XML schema	2.1	31/05/2012
[WSDL]	Datex II v2.0 Push/Pull	2.0	21/01/2011
[Exchange PSM]	DATEX II v2.0 Exchange Platform Specific Model	2.0	30/06/2011
[Schema generation tool]	DATEX II v2.1 XML schema generation tool guide	2.1	31/05/2012
[Extension guideline]	DATEX II v2.1 Extension guideline	2.1	31/05/2012
[Profiling guideline]	DATEX II v2.1 Profiling guideline	2.1	31/05/2012
[Exchange PIM]	DATEX II v1.0 Exchange Platform Independent Model	1.01	08/02/2005

DATEX II in general

2. DATEX II in General

2.1. Background

Delivering European Transport Policy in line with the White Paper issued by the European Commission requires co-ordination of traffic management and development of seamless pan European services. With the aim to support sustainable mobility in Europe, the European Commission has been supporting the development of information exchange mainly between the actors of the road traffic management domain for a number of years. In the road sector, the DATEX standard was developed for information exchange between traffic management centres and constitutes the reference for applications that have been developed in the last 10 years. With DATEX II the DG TREN now also pushes the door wide open for actors from the wider traffic and travel information sector.

Much investment has been made in Europe both, in traffic control and information centres over the last decade and also in a quantum shift in the monitoring of the Trans European Network. This is in line with delivering the objectives of the Tempo projects for safer, better-informed travellers.

Collecting information is only part of the story – to make the most of the investment data needs to be exchanged both with other centres and, in a more recent development, with those developing pan-European services provided directly to road users.

DATEX was designed and developed as a traffic and travel data exchange mechanism by a European task force set up to standardise the interface between traffic control and information centres. It has been the reference for applications that have been developed and implemented in Europe. The existing DATEX network consists of 50 to 60 operational nodes organised in different network and node types throughout Europe. The majority of nodes are used for national exchange of data, but some of them support international exchange.

The DATEX technical documents namely the data dictionary (pre-standard ENV13106:2000) and the DATEX-net specifications for data exchange (pre-standard ENV13777:2000) needed to evolve to reflect technological evolutions, the experience gained in data exchange implementations that were achieved between the European countries and the new needs that have been identified by the market.

Alongside the DATEX pre-standards, a Data Exchange Memorandum of Understanding (DATEX MoU) covering international exchange of traffic data was formally established in October 1997. The MoU confirmed in a formal manner that the development of international traffic data exchange would be based on the DATEX technical specifications, and it established an organisational framework that enabled users to influence and participate in the developments. Different organisations were created under the umbrella of this MoU (**Supervisory Management Committee, Technical Committee and User Forum).**

The signatories of the DATEX MoU decided to work on a revised MoU which is more focused on the availability of traffic and travel data to third parties.

It should be noted that many of the original signatories were participants in the Euro-Regional projects which form the EU deployment programme for ITS, know as the TEMPO Programme[1], involving more than 80 organisations from 14 Member States and three neighbouring countries.

The development of DATEX II was begun in late 2003 and has been supported and partially funded by the European Commission who see it as playing a fundamental role in the ITS domain within European states. This role now extends from traffic control centre / road authority usage to include all types of service provider usage in the ITS domain. Its data content domain is also now extended from the trunk / motorway / TERN road network to include urban network information. Thus DATEX II is aimed at a very wide user base which is far broader than that of the original DATEX specifications.

The original DATEX specifications suffered from a number of shortcomings which made it unlikely to achieve “plug and play” interoperability between DATEX nodes from different manufacturers. Updating the technology, addressing the interoperability issues and the latest stakeholder requirements were the key drivers in the development of DATEX II. DATEX II was not intended to be a rigid set of specifications, but rather one that allowed a degree of choice and one that was able to evolve to allow stakeholders to exchange additional new types of information in the future. However, interoperability between disparate DATEX II systems was still given a high priority.

A first version (1.0) was then produced at the end of 2006 and was then quickly disseminated among countries. The corresponding implementations raised a number of mistakes and requests for

[1] EU funding programme 2001-2006 for the deployment of ITS (**T**rans-**E**uropean Intelligent Transport Syste**m**s **P**r**o**jects). The TEMPO programme has been followed by the new multi-annual programme called EasyWay to the years 2007 – 2013. It is funded by the European Commission. The programme is managed by the Directorate General for Mobility and Transport..

change. After more than one year of work, a first proposal for the version 2.0 was issued in July 2009. In parallel, CEN TC278 circulated the first standard parts for comments. The comment resolution as well as the integration of the work carried out in TG entailed producing a new stable version called “2.0”, basis of the technical specifications issued by CEN (CEN/TS 16157:2011). This version called “2.1” results from the integration of an approved extension (“Parking Information Publication”), the correction of some bugs and improvements brought to the schema generation tool.

2.2. Important features

2.2.1. Platform Independent data model

The DATEX II development focussed on the production of a set of reference documents which make a very clear distinction between data (content) and exchange mechanisms. The main part of the work has been focused on the subject of Traffic and Travel Data Models.

As shown in figure 1, the technical description provides a very clear distinction between platform independent modelling aspects (PIM) and platform specific modelling aspects (PSM). PIM aspects are described in UML.

In each case, a clear distinction between the data modelling (referring to traffic domain) and the data exchange specifications (referring to information and communication technology) were introduced.

	Modelling of data (traffic domain)	Modelling of exchange (ICT domain)
Platform independent models (PIM)	X	X
Platform specific models (PSM)	X	X

Figure 1 – aspects to distinguish in the new descriptions

This approach has been chosen, as it provides several advantages:

The separation of these aspects provides better understanding of the standard to the users and makes it easier to apply.

The platform independent aspects can be considered as more stable than the platform (and technology) specific aspects.

The project has produced explicit models in Unified Modelling Language (UML), which define the contents of DATEX Traffic and Travel publications independent of the exchange mechanism or implementation technology.

The former DATEX specification was a ‘closed’ standard, i.e. a potential user could use the data concepts provided by the Data Dictionary, but if the application required data concepts that could not be found in the Data Dictionary, there was no way for this user to extend the data model without breaking the standard. Many potential users thus ignored DATEX entirely and produced non-interoperable solutions. It is one of the major requirements for the evolution of DATEX to overcome this problem.

2.2.2. Data model levels: A, B, C

The “Level A” data model

DATEX II delivers a **[data model]** also called “Level A data model” which is the result of studies on data which are shared by a lot of users in Europe.

Nevertheless, there will be situations where data concepts required by a particular user are missing in the Data Dictionary, for example because they only make sense in a National context. In this case, these users are expected to provide an extension to the model (named “level B”) that provides the missing concepts. Users are allowed to apply a limited set of well-defined UML mechanisms for these level B extensions, which then still maintain technical interoperability with standard DATEX II systems. This means that standard (i.e. level A) compliant systems will still be able to process publications generated from an extended model, of course without being able to process the extension content. Specialised clients can process the full content – including the extension – but of course can also process standard (i.e. level A) publications.

The main principle of level B is thus that users find the level A model appropriate for a large part of their publications. This may not be the case if entirely new concepts outside the scope of the level A model are introduced. In this scenario, DATEX II users are still expected to apply the modelling rules and the UML profile described in this report, which will provide them with a basic level of interoperability. These models are denoted as "level C". Implementations that provide publications according to level C extensions cannot expect to be interoperable with standard level A systems.

The extended model – "Level B" extension mechanisms

Although the "A" model is rich, over time there will always be cases where new applications, both at national and international level, will want to add additional concepts and attributes to the existing models. To cater for this future proofing aspect of the modelling it is desirable to have a formal mechanism by which the "A" model can be extended.

For these new applications requiring extensions to the "A" model, the concept of Level B compliance has been created. This will allow development of specific models that will enrich the "A" model with additional, application specific information.

These models/applications will remain interoperable with "A" model compliant suppliers/consumers: they can exchange objects structured according to these enriched models.

Suppliers/consumers that want to make full use of the information defined by these extensions to the "A" model will need additional software or metadata driven software to handle the additional application specific information. An additional guideline [**Extension guideline**] provides editors with useful information on how to create compliant level B extensions.

A registration process for level B model extensions is available on the DATEX II website (www.datex2.eu), under the authority of the DATEX TG. Once an extension achieves a major degree of consensus it would become a candidate to be absorbed into a new version of the formal DATEX II level A.

Using the DATEX II concepts within different contents – "Level C" Users and Models

After consideration of Level A and Level B compliance rules some users within the ITS domain may still find that there is no way that their specific data models can be accommodated. They are just too different from the Level A model or else cover completely different contents. That's why the concept of "level C" has been created.

Level C implementations are to be considered as not compliant with the DATEX II Level A/B content models. However, they are to be compliant in all other aspects of the DATEX II specifications.

Obviously these Level C compliant systems would not be interoperable with Level A compliant systems, but at least they would use common modelling rules and common exchange protocols.

This would allow opportunities for the exchange of ideas and modelled concepts which may in the future lead to common model elements facilitated via some sort of model registration process. It will also permit many helpful software tools to be used with Level C compliant contents.

2.2.3. Data dictionary

The previous DATEX standard provided definitions which were readable by traffic engineers and IT experts alike. As the definitions for the data content in DATEX II are only stored within the UML model, which are not simply accessible for reading by non-IT professionals, a readable data dictionary is available via a software tool which automatically extracts the definitions from the UML model. The dictionary is accessible via the DATEX II web site (http://www.datex2.eu).

2.2.4. XML schema

Today there is a growing market and use of XML for information exchange. A fundamental part of this approach is the XML Schema. Based on a data model and a data dictionary the XML Schema is designed to fit a certain area of applications. The schema is the tool to understand the content of the exchanged data.

A conversion tool has been developed to convert the UML DATEX II data model into an **[XML schema].** The UML model is first exported from the UML modelling tool into an XMI file which the conversion tool then converts to an XML schema. The XML schema is used for development of real exchanges, by software developers.

XMI (XML Metadata Interchange) is an OMG standard for exchanging metadata information via XML. It can be used for any metadata whose metamodel can be expressed in MOF (Meta-Object Facility). The most common use of XMI is as an interchange format for UML models, although it can also be used for serialization of models of other languages (metamodels).

The following figure shows the workflow for an automated conversion process.

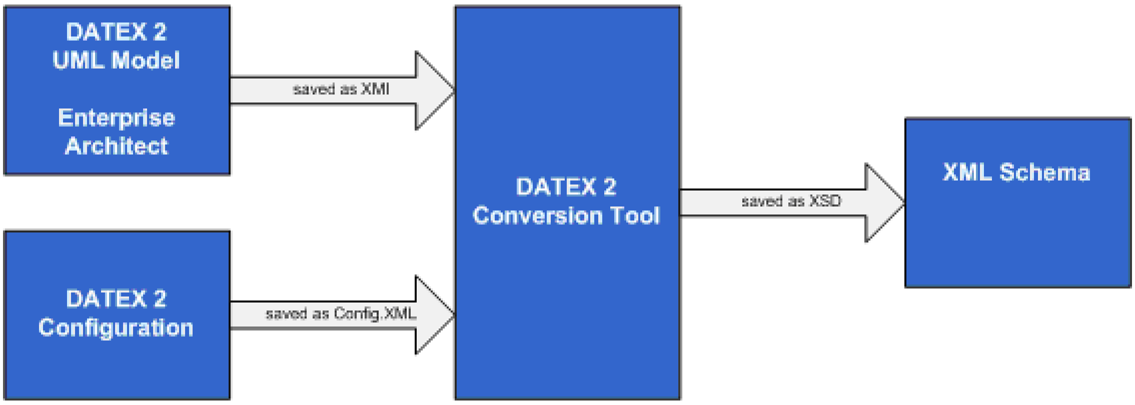


Figure 2 – UML -> XSD conversion workflow

2.2.5. Exchange mechanisms

Generally speaking, DATEX II offers a push and a pull mode for information exchange. The push mode (as in DATEX) allows the supplier to send information to the client while the pull mode allows the client to request the download of information from the supplier's systems. In detail, DATEX II provides the exchange mechanisms as described below.

At a high description level document for details), without considering the notion of technical platforms, data exchange between a supplier and its client(s) can be accomplished by three main operating modes:

- Operating Mode 1 – Publisher Push on occurrence
  - data delivery initiated by the publisher every time data is changed
- Operating Mode 2 – Publisher Push periodic
  - data delivery initiated by the publisher on a cyclic time basis
- Operating Mode 3 – Client Pull
  - data delivery initiated by the Client, where data is returned as a response.

Each Operating Mode can be both on- and offline.

For the "Client Pull" operating mode, two implementation profiles have been defined for implementing this operating mode over the Internet: by direct use of the HTTP/1.1 protocol or via Web Services over HTTP.

For the "Supplier Push" operating modes, one platform has been defined using Web Services over HTTP.

The common corresponding document, describing all operating modes and both profiles for Client Pull as well as their interoperability, is [**Exchange PSM**]. This document shall be used to have information on how to proceed with webservices and [**Wsdl**]. The background information is provided with [**Exchange PIM**]. This outdated document should be totally rewritten soon.

PSM exchange documents have been designed to be independent from the exchanged content (the payload). These documents can be studied without knowing the details of the UML DATEX II data model.

2.2.6. DATEX II Profiles

DATEX II allows exchange for several kinds of data via several operating modes.

Not all of those operating modes have necessarily to be implemented in every DATEX II system and not all data content need be implemented. Thus DATEX II allows the implementation of profiles.

“Profiling” aims to define a customised subset of options offered by a standard for a particular need.

What is a Profile?

A DATEX II system is composed of different publications which can be delivered with different operating modes. Each DATEX II system builder chooses to implement the subset of couple publications - operating modes he needs. This subset is called a « DATEX II profile ».

The need is to have profiles and options that allow DATEX II users to customize their implementations in order to provide more or less functionalities/facilities as necessary and not to be forced into implementing all the features.

DATEX II allows every user to define a profile according to his own requirements whilst keeping interoperability on common parts (publications, operating modes) with other users.

What must the stakeholder choose?

Because different needs for different use cases of DATEX II may lead to the definition of different profiles, this step will require close stakeholders’ involvement to elicit their requirements.

Moreover, profiling will require assessment of the cost/benefit trade-off, in particular of:

- standard features/services;
- implementation platforms; and
- level of service that must be achieved.

Stakeholders need to provide their own perspective which will influence the main choices to be made concerning:

- the list of publications to be exchanged,
- the operating modes to be implemented,

and sometimes also, options (for instance option “Web Service”/”Simple http” in the Pull operating mode 3).

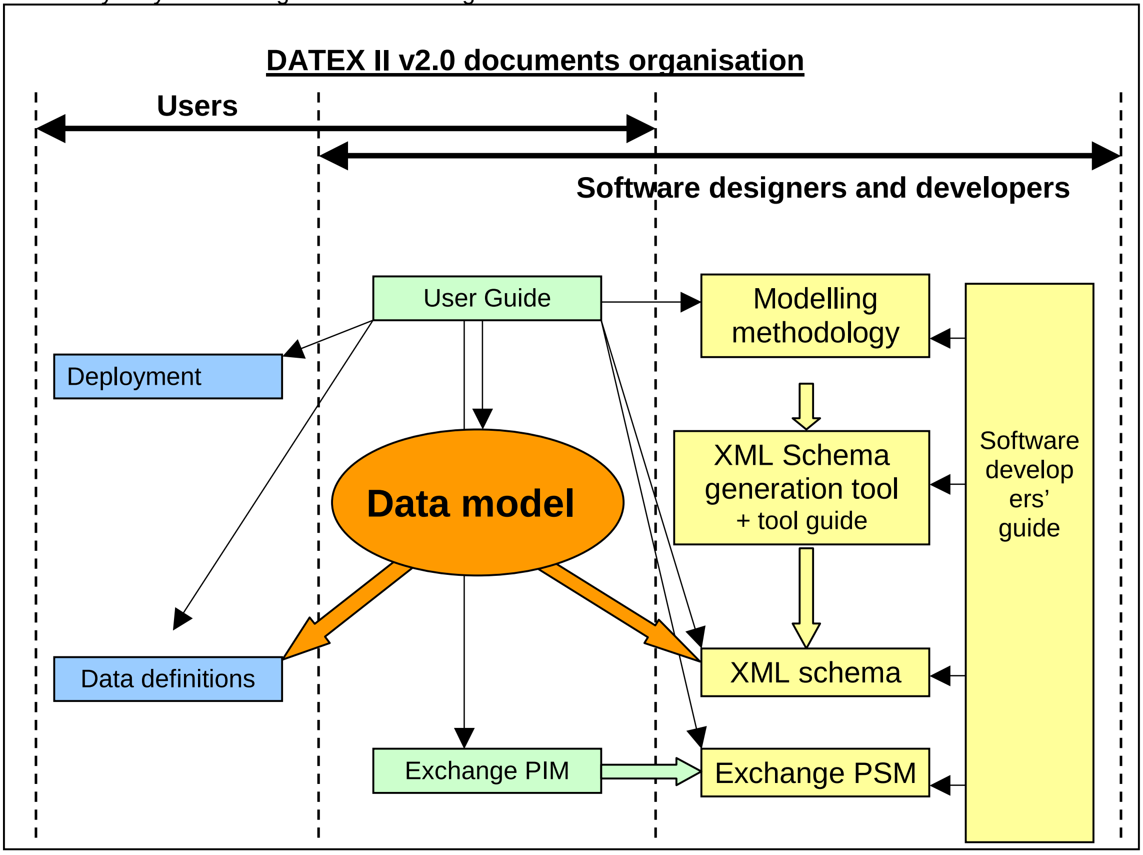
The following document [**Profiling guideline**] contains more information on how to profile the data model and obtain the corresponding subschemas.

2.3. DATEX II documents organisation

The organisation of the DATEX II documents is shown in the following diagram.

A documents priority depends on the reader's type (Users or software developer).

Everybody should begin with this user guide.


DATEX II v2.0 documents organisation
Users
Software designers and developers
User Guide
Deployment
Data model
Data definitions
Exchange PIM
Modelling methodology
XML Schema generation tool + tool guide
XML schema
Exchange PSM
Software developers' guide

The DATEX II v2.1 Reference Set is composed of:

- Data model
- Modelling methodology
- XML schema generation tool
- XML schema
- Exchange PSM

The informative documents which are used to understand DATEX II and make main decisions are:

- User Guide
- Data definitions (dictionary)
- Software developers Guide
- Extension guideline
- Profiling guideline

Exchanged data

3. Exchanged data

3.1. Introduction

Before diving into the UML model, this chapter presents the exchanged data.

Exchanged data are made with basic elements which are available inside publications.

3.2. Basic elements

Information exchanged with DATEX II systems is composed of different **basic elements**:

- Road and traffic related events (called “Traffic elements”)
- Operator actions
- Impacts
- Non-road event information
- Elaborated data (derived/computed data, e.g. travel times, traffic status)
- Measured data (direct measurement data from equipments or outstations, e.g. traffic and weather measurements)
- Messages displayed on Variable Message Signs (VMS).

In addition there are also Predefined Locations, VMS Table and Measurement Site Table information exchanged. They are not part of the basic elements, but are required if the corresponding information in the basic elements is to be understood by a client.

Road and traffic related events (called “Traffic elements”)

These are all events which are not initiated by the traffic operator and force him to undertake (re)actions. They are classified in 6 main categories:

- Abnormal traffic (long queues, stop and go, …)
- Accidents
- Obstructions:
  - animal presence,
  - vehicle presence,
  - obstructions due to environment (avalanches, flooding, fallen trees, rock falls, …),
  - obstructions due to infrastructure (fallen power cables, …)
  - other obstructions including people
- Activities (public event, disturbance, …)
- Incidents on equipments or systems (Variable message sign out of order, tunnel ventilation not working, emergency telephone not working, …)
- Conditions: driving conditions related to weather (ice, snow, …) or not (oil, …), conditions related to environment (precipitation, wind, …), …

Operator actions

Operator actions are classified in 4 main categories:

- Network management: road closure, alternate traffic, contraflow …
  - traffic control: rerouting, temporary limits
- Roadworks: resurfacing, salting, grass cutting …
- Roadside assistance: vehicle repair, helicopter rescue, food delivery …
- Sign settings: This refers to a VMS message.

Impacts

It contains, in particular, information on lane availability and on delays (in seconds, in time range or globally).

Non-road event information

It concerns information about events which are not directly on the road: transit service information, road operator service disruption, car parks.

Elaborated data

These sets of data are normally derived on a periodic basis by the Traffic Centre from input data for specified locations:

- travel times: elaborated time, free flow time, normally expected time
- traffic status = attribute with 5 possible values:
    - free flow,
    - heavy,
    - congested,
    - impossible,
    - unknown
- traffic values (normally published as measured data, but can be derived on a periodic basis and published as elaborated data): flow, speed, headway, concentration and individual vehicle measurements.
- weather values (normally published as measured data, but can be derived on a periodic basis and published as elaborated data): precipitation, wind, temperature, pollution, road surface condition and visibility.

They can be forecast values.

Measured data

These data sets are normally derived from direct inputs from outstations or equipments at specific measurement sites (e.g. loop detection sites or weather stations) which are received on a regular (normally frequent) basis:

- traffic values: flow, speed, headway, concentration and individual vehicle measurements.
- weather values: precipitation, wind, temperature, pollution, road surface condition and visibility.
- travel times (normally published as elaborated data, but direct outstation values can be published as measured data): elaborated time, free flow time, normally expected time
- traffic status (normally published as elaborated data, but direct outstation derived values can be published as measured data) = attribute with 5 possible values (see Elaborated data above).

VMS messages

These data sets include different possible messages according to different technologies, including textual messages, pictograms or combinations as well as allowing for full matrix VMS. They are completed by some information about equipment status and position.

3.3. Publication of basic elements

The previous basic elements can be exchanged individually or grouped. For these exchanges, the notion of **publication** is used. There are 5 main publications:

- Situation publication
- Elaborated data publication
- Measured data publication
- Traffic View publication
- VMS publication

The following table presents what can be exchanged with each publication.

	Situation publication	Elaborated Data publication	Measured Data publication	Traffic View publication	VMS publication
Traffic Elements	X			X	
Operator Actions	X			X	
Impacts	X				
Non-road event information	X				
Traffic Status		X	X[2]	X	
Travel Times		X	X[2]	X	
Traffic Values		X[2]	X	X	
Weather Values		X[2]	X	X	
VMS messages	X				X

Situation publication

A situation publication can contain several different situations.

A **situation** represents a traffic/travel situation comprising one or more traffic/travel circumstances which are linked by one or more causal relationships and which apply to related locations. Each traffic/travel circumstance is represented by a Situation Record.

A **situation record** is one element of a situation. It is characterized by values at a given time, defining one version of this element. When these values change, a new version is created. One situation record can be:

- An road or traffic related event (traffic element),
- An operator action,
- A non-road event information,

and can contain:

- Impact details.

Elaborated data publication

This publication is used to send periodically elaborated data derived by a Traffic Centre relating to specified locations. Locations may be explicitly defined in the publication or, more simply, may be referred to by references to predefined (group of) locations which have been exchanged via the "predefined locations" publication.

Measured data publication

This publication is used to send periodically measured data which has been derived from equipment at specific measurement sites, where each site is identified by reference to an entry in a predefined measurement site table. The measurement site table can be exchanged via the "measurement site table" publication and provides for each site the details of its location and parameters associated with the different types of measurements that can be made at the site.

Each set of measurements from a site is ordered (i.e. indexed), where each ordered measurement might be of a different type. The order (or indexing) of these measurements for each site within the Measured Data publication must correspond with the ordered (indexed) definition of the measurement in the Measurement Site Table publication for the particular site.

VMS publication

This publication is used to send periodically messages elaborated by a Traffic Control Centre relating to a set of display equipments among a road network. Static VMS characteristics may be

[2] Possible but not the most adapted

explicitly defined in the publication (for mobile VMS e.g.) or, more simply, may be referred to by references to predefined VMS installations which have already been exchanged via the “VMS table” publication.

Traffic View publication

The traffic view is also a new DATEX II concept.

A traffic view is a snapshot of what happens on one itinerary, in one direction at a given time.

A traffic view is autonomous: it contains all necessary information to understand what is happening on this itinerary at the specific point in time.

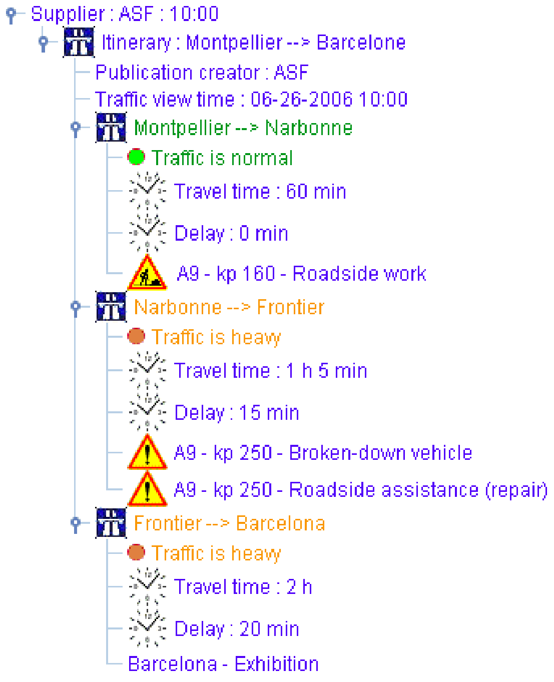
There is no historical management. There is no need to manage a link failure and then recovery because, by definition, traffic view n° J cancels and replaces traffic view n° J-1.

A traffic view is organised in “oriented road sections”. Each section can contain:

- Elaborated data (traffic status, travel time, traffic and weather measurements)
- Road or traffic related events (if any)
- Operators actions (if any)

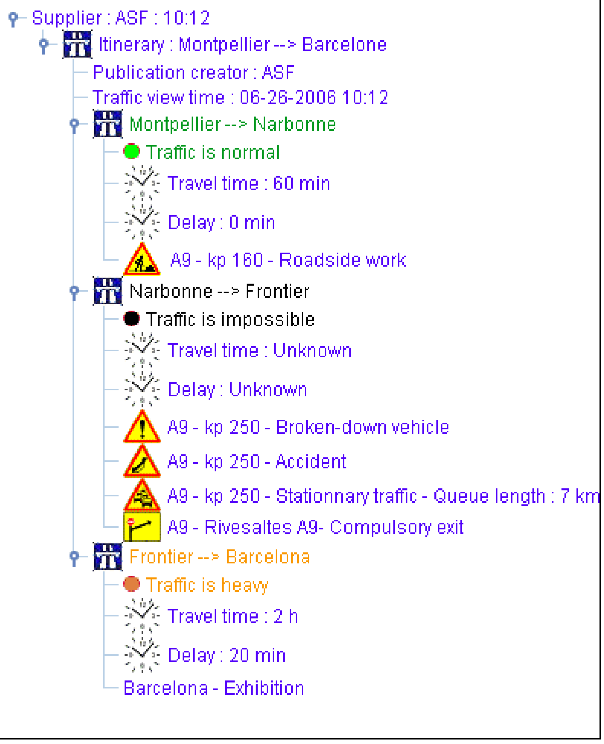
Example of traffic view display[3] (snapshot n° J and snapshot n° J+2)

Traffic view n° J

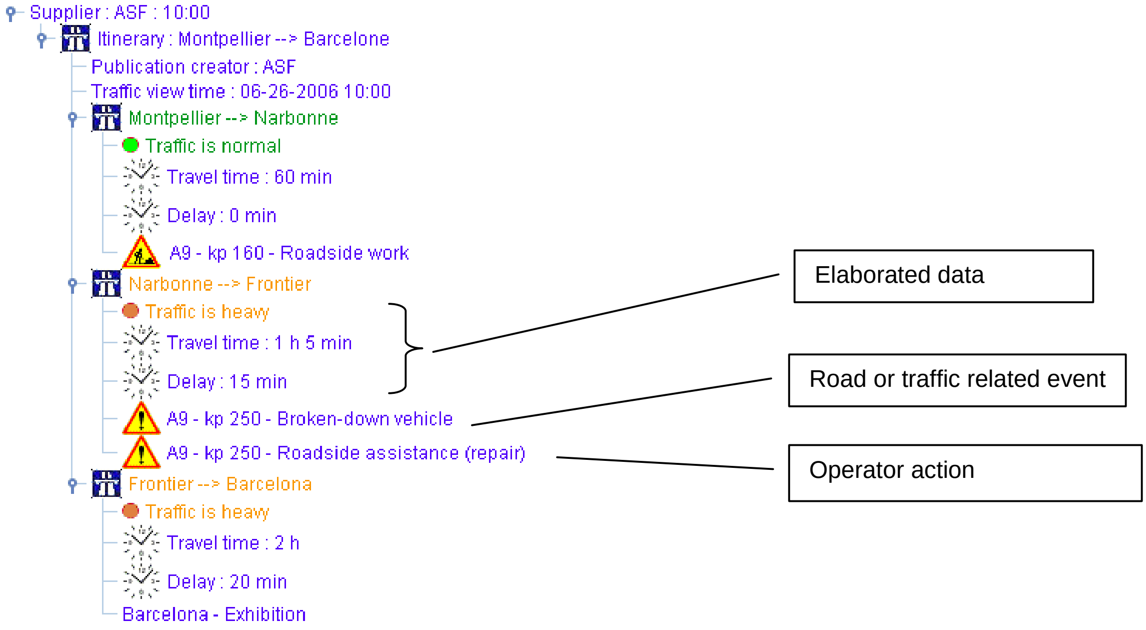


- Supplier : ASF : 10:00
  - Itinerary : Montpellier --> Barcelone
    - Publication creator : ASF
    - Traffic view time : 06-26-2006 10:00
    - Montpellier --> Narbonne
      - ● Traffic is normal
      - Travel time : 60 min
      - Delay : 0 min
      - A9 - kp 160 - Roadside work
    - Narbonne --> Frontier
      - ● Traffic is heavy
      - Travel time : 1 h 5 min
      - Delay : 15 min
      - A9 - kp 250 - Broken-down vehicle
      - A9 - kp 250 - Roadside assistance (repair)
    - Frontier --> Barcelona
      - ● Traffic is heavy
      - Travel time : 2 h
      - Delay : 20 min
    - Barcelona - Exhibition

Traffic view n° J+2



- Supplier : ASF : 10:12
  - Itinerary : Montpellier --> Barcelone
    - Publication creator : ASF
    - Traffic view time : 06-26-2006 10:12
    - Montpellier --> Narbonne
      - ● Traffic is normal
      - Travel time : 60 min
      - Delay : 0 min
      - A9 - kp 160 - Roadside work
    - Narbonne --> Frontier
      - ● Traffic is impossible
      - Travel time : Unknown
      - Delay : Unknown
      - A9 - kp 250 - Broken-down vehicle
      - A9 - kp 250 - Accident
      - A9 - kp 250 - Stationnary traffic - Queue length : 7 km
      - A9 - Rivesaltes A9- Compulsory exit
    - Frontier --> Barcelona
      - ● Traffic is heavy
      - Travel time : 2 h
      - Delay : 20 min
    - Barcelona - Exhibition



- Supplier : ASF : 10:00
  - Itinerary : Montpellier --> Barcelone
    - Publication creator : ASF
    - Traffic view time : 06-26-2006 10:00
    - Montpellier --> Narbonne
      - ● Traffic is normal
      - Travel time : 60 min
      - Delay : 0 min
      - A9 - kp 160 - Roadside work
    - Narbonne --> Frontier
      - ● Traffic is heavy
      - Travel time : 1 h 5 min
      - Delay : 15 min
      - A9 - kp 250 - Broken-down vehicle
      - A9 - kp 250 - Roadside assistance (repair)
    - Frontier --> Barcelona
      - ● Traffic is heavy
      - Travel time : 2 h
      - Delay : 20 min
    - Barcelona - Exhibition

[3] This display is just an example in a tree representation of a traffic view and does not imply any specific traffic view requirements. Representations of traffic views can be done in many other ways and is open to implementation design.

UML data model content

4. UML data model content

4.1. Objectives

This chapter is intended to assist readers in understanding the structure and content of the DATEX II UML model.

When reading this document, the Data Dictionary tool and documentation which groups all definitions (classes, attributes, enumeration values ...) may also be of help.

4.2. Prerequisite

The UML model is available on the DATEX web site: http://www.datex2.eu in an Enterprise Architect .eap file.

The reader must have one Enterprise Architect tool version installed in order to read the model (refer Sparx Systems web site: http://www.sparxsystems.com).

The UML model may also be directly navigated on the website (html version)

The reader should be familiar with UML modelling.

The UML basic class diagram elements used in the model are presented briefly in Annex 1.

4.3. Modelling methodology

For more details of the UML features used in the model please refer to document [**Modelling methodology**].

4.4. Data model structure

The model comprises several diagrams:

- Analysis diagrams,
- Dynamic diagrams,
- Functional diagrams,
- Logical diagrams.

Analysis, dynamic and functional diagrams

These diagrams were designed when the Platform Independent Model for exchange was built: see document [**Exchange PIM**].

To understand these diagrams, it is the best to read this document where they have been included. No further explanation on these is provided here.

4.5. Logical diagrams structure

The logical diagrams are divided in 4 packages:

- **Exchange**: contains exchange diagram and exchange enumeration definitions
- **PayloadPublication**: containing description of publications data to be exchanged
- **General**: containing “general classes” : Data types, Location classes, Payload enumerations, reusable classes
- **Management**: contains, for this document version, situation management in operating mode “Publisher Push on occurrence”
- **Extension**: is a placeholder for containing the elements composing possible extension.

The logical diagram hierarchy is as follows:

- D2LogicalModel
  - Exchange
    - ExchangeEnumerations
  - PayloadPublication
    - SituationPublication
      - SituationRecord
        - Impact
        - Validity
          - TimePeriodOfDay
        - TrafficElement
          - Obstruction
            - Vehicle
          - Accident
            - Vehicle
            - VehicleCharacteristics
          - Activity
          - Conditions
        - OperatorAction
          - Roadworks
          - NetworkManagement
            - VehicleCharacteristics
        - NonRoadEventInformation
    - ElaboratedDataPublication
      - ElaboratedData
        - Validity
          - TimePeriodOfDay
        - BasicData
          - TrafficStatus
          - TravelTimeData
          - TrafficData
            - VehicleCharacteristics
            - TrafficHeadway
            - TrafficFlow
            - TrafficSpeed
            - TrafficConcentration
          - WeatherData
            - Visibility
            - Humidity
            - Pollution
            - PrecipitationDetail
            - RoadSurfaceConditionMeasurements
            - Temperature
            - Wind

```
            GroupOfLocations
PredefinedLocationsPublication
TrafficViewPublication
    ElaboratedData
    TrafficElement
    OperatorAction

MeasuredDataPublication
    BasicData
            TrafficStatus
            TravelTimeData
            TrafficData
                VehicleCharacteristics
            WeatherData
                Visibility
                Humidity
                Pollution
                PrecipitationDetail
                RoadSurfaceConditionMeasurements
                Temperature
                Wind
            GroupOfLocations
MeasurementSiteTablePublication
    VehicleCharacteristics
    GroupOfLocations

VmsPublication
    VmsUnit
        VMSMessage
        Location
VmsTablePublication

GroupOfLocations
    Point
        PointAlongLinearElement
        AlertCMethod2Point
        AlertCMethod4Point
        TPEGPointLocation
    Linear
        LinearWithinLinearElement
        AlertCMethod2Linear
        AlertCMethod4Linear
        AlertCLinearByCode
        TPEGLinearLocation
    Area
```

AlertCArea

TPEGAreaLocation

Each diagram is presented in the following paragraphs.

4.6. D2LogicalModel

At the top level the D2LogicalModel comprises two components:

- Exchange
- PayloadPublication

The Exchange class is used to exchange parameters associated with the data delivery.

The Payload sub-model comprises the various publications which contain primarily traffic and travel related data content. Some publications are also used for the exchange of static or near static tables, such as those used for location referencing or for locating static measurement equipment on a road network. However, the Payload sub-model is where all the real traffic and travel information is located.

4.7. Exchange

The Supplier Identification is mandatory (2 letters country enumeration (and “other”) plus a national identifier in that country).

When a subscription has been established between the supplier and the client(s), the corresponding class can be filled.

If the Supplier has a catalogue and if data delivery refers to catalogue references, these references can be indicated. Note that the exchange of catalogues in this version of DATEX II is not supported on-line.

If a filter has been used before data publication, filter characteristics can be indicated. Note that the exchange of filters in this version of DATEX II is not supported on-line.

The target class can be used when the payload concerns only one destination address.

4.8. PayloadPublication

There are 5 main publications:

- Situation publication
- Elaborated data publication
- Measured data publication
- Traffic View publication
- VMS publication

There are 3 other publications, which define references used by the previous publications.

Each publication must have:

- its creation time “PublicationTime”,
- the language to be used by default,
- a publication creator which is an international identifier: a two-letter country enumeration (and “other”) plus a national identifier in that country. This creator may be different of the publication supplier.

The following paragraphs detail each publication.

The groups of locations, which are used by every publication, are detailed in a dedicated paragraph, after the description of the different publications.

4.9. Situation publication

4.9.1. Main features

A situation publication can group several situations.

Situation definition: An identifiable instance of a traffic/travel situation comprising one or more traffic/travel circumstances which are linked by one or more causal relationships. Each traffic/travel circumstance is represented by a Situation Record.

Warning: each situation has a unique versioned identifier. This versioned identifier comprises a part established when the situation is first created in a DATEX II system database and kepts within that system for all its life. The second part ("version") is established at every new version of the situation. Such mechanism allows having several versions of the same situation in the same publication

The situation severity on traffic can be indicated: "overallSeverity".

If this situation relates to another situation, its reference can be indicated: "relatedSituation" (this reference is the "versioned identifier" of the corresponding situation object).

With this publication, attributes can be given in the InformationHeader which provide clients with information management details (area of interest, confidentiality, urgency, information status) relating to this situation.

A situation may be composed of one or more situation records.

Special case: when the Supplier sends SituationPublication to Clients in Operating Mode 1 "Publisher Push on occurrence", the situation management has to follow different rules. This is explained in paragraph 4.9.6: Special case: Publisher Push on occurrence.

4.9.2. Situation record

Situation record definition: An identifiable instance of a single record/element within a situation.

This class is abstract.

Each situation record has a **unique versioned identifier.** This first part of this identifier is established when the situation record is first created in the DATEX II system database. It is completed by a version part. The first part is kept within that system for all its life. The subsequent versions of this situation record must differ by their version part of the identifier. If the situation record is forwarded down a supply chain, each DATEX II system within that chain will assign its own unique identifier to the record when it first creates a corresponding record in its database, but the (compound) identifier created by the original supplier in the chain may optionally be forwarded in the record's **situationRecordCreationReference attribute**.

It is located with **one group of locations** (refer "location" paragraph).

It can have **one cause**.

It can have **one impact** on the traffic (when the record type is "TrafficElement" or "OperatorAction"), qualified with:

- Indications on delays (code, value range, value)
- Other details (capacity remaining, effect on lanes, restriction type)..

Source information can be given (country, identifier, name, type, if reliable or not)

A situation record has a **validity** composed of:

- An validity status which defines if the record has an "active" or "suspended" status, or this status is defined according to periods of time .
- One mandatory global first start time "overallStartTime"
- One optional global real end time "overallEndTime"
- One or several validity periods, each having a start date and an end date, for which the situation record is active/valid, detailed with hours/days/weeks/months
- One or several exception periods, each having a start date and an end date, for which the situation record is not active/valid, detailed with hours/days/weeks/months
- The optional "overrunning" means the activity/action is still in progress, overrunning its planned duration as indicated in a previous version of the record.

Remarks for **forecasts**:

- When a record is created, with a start time in the future, it means it is a « forecast » and therefore for traffic elements, probalityOfOccurrence should never have the value « certain ».
- When this start time arrives, the Supplier should update or end this information if it has not really occurred. If a supplier does not provide updated information at this time a client **can not** assume that the traffic element is now active, only that it was predicted to be active now.

There are 3 main categories of situation records:

- **Traffic element**: An event which is not planned by the traffic operator, which is affecting, or has the potential to affect traffic flow (note this does includes events planned by external organisations e.g. exhibitions, sports events etc.).
- **Operator action**: Actions that a traffic operator can decide to implement to prevent or help correct dangerous or poor driving conditions, including maintenance of the road infrastructure.
- **Non-road event information**: Information about an event which is not on the road, but which may influence the behaviour of drivers and hence the characteristics of the traffic flow.

These 3 categories are detailed in the following paragraphs.

4.9.3. Traffic element

This class is abstract.

There are 6 kinds of traffic elements:

- Obstruction,
- Abnormal traffic,
- Accident,
- Equipment or system fault,
- Activities,
- Conditions

Obstruction: Any stationary or moving obstacle of a physical nature (e.g. obstacles or vehicles from an earlier accident, shed loads on carriageway, rock fall, abnormal or dangerous loads, or animals etc.) which could disrupt or endanger traffic.

This class is abstract.

For each obstruction type, the number of obstructions can be given.

For each obstruction type, details can be given of:

- Mobility type: mobile/stationary,

There are 5 kinds of obstructions:

- Animal presence (alive or not, presence type)
- Environmental obstruction (depth, type),
- Infrastructure damage obstruction (type),
- General obstruction (type),
- Vehicle obstruction (type + individual vehicles characteristics)

Abnormal traffic: A traffic condition which is not normal.

Details can be given: type, number of vehicles waiting, queue length, relative traffic flow, traffic flow characteristics, traffic trend type.

Accident: Accidents are events in which one or more vehicles lose control and do not recover. They include collisions between vehicle(s) or other road user(s), between vehicle(s) and any obstacle(s), or they result from a vehicle running off the road.

Details can be given:

- cause type,
- accident type,
- overview of people involved (number, injury status, involvement role, type),
- overview of vehicles involved per type (number, status, type, usage),
- details of involved vehicles (type + individual vehicle characteristics)

Equipment or system fault: Equipment or system which is faulty, malfunctioning or not in a fully operational state that may be of interest or concern to road operators and road users.

The type of equipment or system and the type of fault must be given.

Activities: Deliberate human actions external to the traffic stream or roadway which could disrupt traffic.

Details must be given in term of: types of authority operations / disturbance / public event, and possibly in term of mobility(mobile/stationary).

Conditions: Any conditions which have the potential to degrade normal driving conditions.

A general indicator can be given with 8 possible values “drivingConditionType”:

- impossible,
- very hazardous,
- hazardous,
- passable with care,
- winter conditions (driving conditions are consistent with those expected in winter),
- normal,
- other,
- unknown.

There are 3 conditions categories:

- road surface conditions that are related to the weather which may affect the driving conditions, such as ice, snow or water,
- road surface conditions that are not related to the weather but which may affect driving conditions, such as mud, oil or loose chippings…,
- environment conditions which may be affecting the driving conditions without being directly linked to the road (precipitation, visibility, pollution, temperature, wind)

4.9.4. Operator action

For each operator action type, details can be given of:

- Origin: internal / external,
- Status: requested, approved, being implemented, implemented, rejected, termination requested, being terminated,
- Considered action plan

There are 4 kinds of operator actions:

- Roadworks,
- Sign setting,
- Network management,
- Roadside assistance.

Roadworks: Highway maintenance, installation and construction activities that may potentially affect traffic operations.

This class is abstract.

The effect on road layout is mandatory.

Details can be given of:

- duration,
- scale: major / medium / minor
- under traffic or not,
- urgent or not,
- mobility type: mobile / stationary,
- construction work type: blasting / construction / demolition, road improvement, road widening,
- road maintenance type (e.g. grass cutting, resurfacing, repair, road marking, salting…,
- subject type of works (e.g. bridge, crash barrier, gantry, road tunnel,…,
- information on associated maintenance vehicles.

Sign setting: Provides information on variable message and the information currently displayed. It uses the class VmsUnit.

Network management: Changes to the configuration or usability of the road network whether by legal order or by operational decisions. It includes road and lane closures, weight and dimensional restrictions, speed limits, vehicle restrictions, contra-flows and rerouting operations.

There are 6 types of network management:

- Rerouting management (type, itinerary description, sign posted or not,…),
- Speed management (type, speed value),
- Road, carriageway or lane management (type, specified lane or carriageway, minimum number of persons in a vehicle required for HOV/car pool lane),
- Winter driving management (type),
- General instructions to road users (type),
- General network management (type of management and type of person that is manually directing traffic in case of manually directed traffic).

Information on whether the network management instruction or the control resulting from a network management action is advisory or mandatory shall be given.

Other details can be given:

- In which traffic direction the traffic management is applicable,
- For which vehicles the traffic management is applicable,
- If the traffic management instruction has been automatically initiated or not,
- The places (defined in generic terms) where the network management applies (e.g. at toll plaza, in galleries, at high altitude, on slip roads, on the crest of hills),
- Vehicles characteristics for which the network management applies,

Roadside assistance: Details of roadside assistance required.

The roadside assistance type can be given: food delivery, helicopter rescue, vehicle repair …

4.9.5. Non-road event information

This class is abstract.

There are 4 kinds of non-road event information:

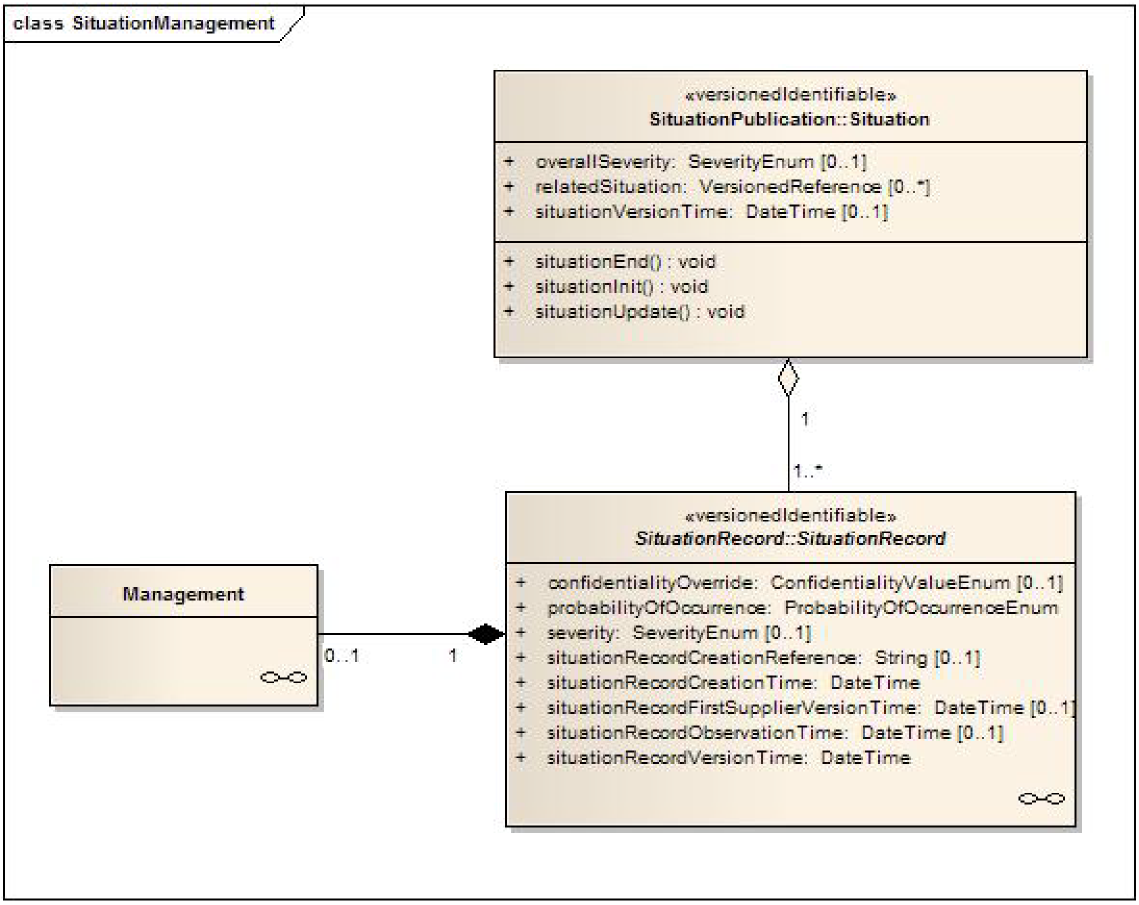
- Availability of transit services and information relating to their departures. This is limited to those transit services which are of direct relevance to road users, e.g. connecting rail or ferry services.
- Service disruption depending of road operator: emergency or information service telephone number out of service, no traffic officer patrol service,
- Service disruption information: fuel shortage, service area closed, some commercial service closed etc.
- Information on car parks: number of vacant spaces, fill rate etc.

4.9.6. Special case: Publisher Push on occurrence

Exchanges between Suppliers and Clients can follow different modes.

For one of these, "Operating mode 1: Publisher Push on occurrence, data delivery is initiated by the publisher, every time data is changed.

In this operating mode, the life cycle of exchanged entities has to be managed (creation, update… end) and follow precise rules. These rules are defined in the **"Management" package** of the UML model and reachable via 2 diagrams:

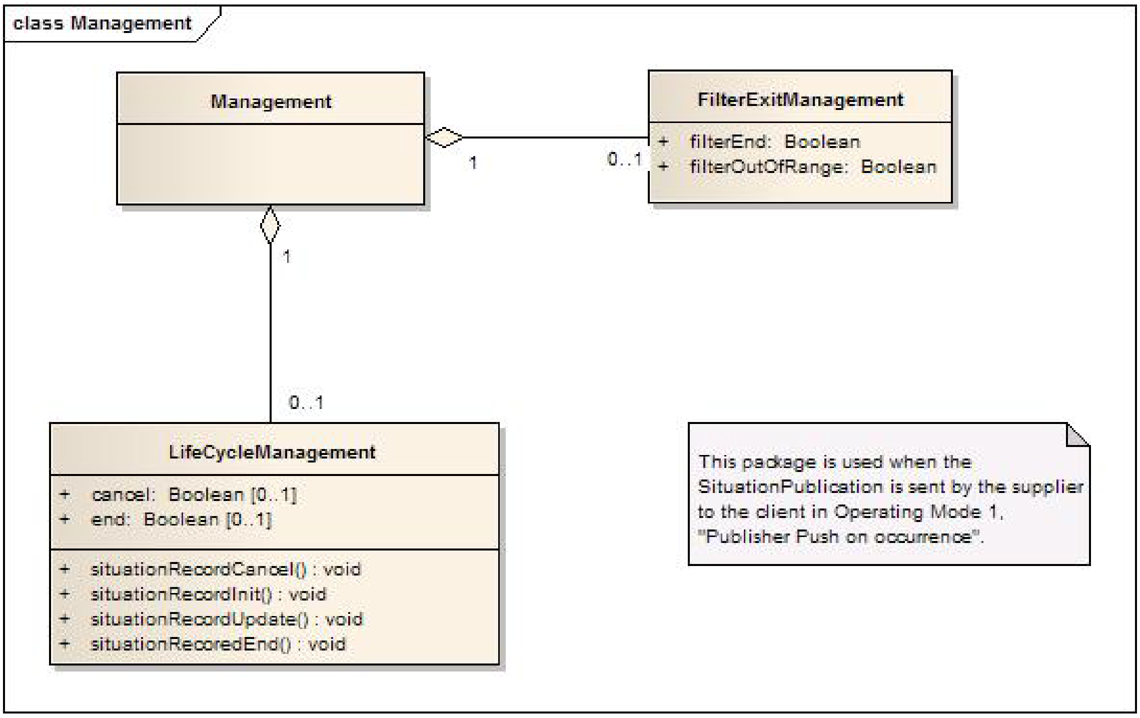

class SituationManagement
«versionedIdentifiable»
SituationPublication::Situation
+ overallSeverity: SeverityEnum [0..1]
+ relatedSituation: VersionedReference [0..*]
+ situationVersionTime: DateTime [0..1]
+ situationEnd() : void
+ situationInit() : void
+ situationUpdate() : void
1
1..*
«versionedIdentifiable»
SituationRecord::SituationRecord
+ confidentialityOverride: ConfidentialityValueEnum [0..1]
+ probabilityOfOccurrence: ProbabilityOfOccurrenceEnum
+ severity: SeverityEnum [0..1]
+ situationRecordCreationReference: String [0..1]
+ situationRecordCreationTime: DateTime
+ situationRecordFirstSupplierVersionTime: DateTime [0..1]
+ situationRecordObservationTime: DateTime [0..1]
+ situationRecordVersionTime: DateTime
Management
0..1
1

- **Situation management** diagram

In this diagram, 3 operations are visible on the Situation class. These operations are to be implemented in a DATEX II client:

- **situationInit**: The situation creation is done automatically upon the reception of a situation record, for the first time, for this situation.
- **situationUpdate**: On updating the state of an existing situation, alterations may be made by means of creating an additional new situation record or by updating the information within an existing situation record.
- **situationEnd**: A situation is ended when all of its situation records have been ended (refer LifeCycleManagement description).
  Remark: a situation can be ended on the Client's side when all of its situation records are not active any more. A situation record is not active when is has been

ended or the validityEndTime has been reached (refer SituationRecord description).


class Management
Management
FilterExitManagement
+ filterEnd: Boolean
+ filterOutOfRange: Boolean
1
0..1
1
0..1
LifeCycleManagement
+ cancel: Boolean [0..1]
+ end: Boolean [0..1]
+ situationRecordCancel() : void
+ situationRecordInit() : void
+ situationRecordUpdate() : void
+ situationRecoredEnd() : void
This package is used when the SituationPublication is sent by the supplier to the client in Operating Mode 1, "Publisher Push on occurrence".

- **Management** diagram

In this diagram:

- 4 operations are visible on the LifeCycleManagement class. These operations are to be implemented in a DATEX II client:
  - **situationRecordInit**: Initialises the first version of the record.
  - **situationRecordUpdate**: Updates situationRecord information.
  - **situationRecordCancel**: A situation record should be cancelled using the "cancel" attribute set to "true". In this case, part or all of the information previously sent for this record was wrong, all information previously sent must be considered as not valid anymore. If the supplier has new valid information, this should be put in the same publication. If the supplier now considers this record's existence as an error, the supplier must end it with the end attribute set to true.
  - **situationRecordEnd**: A situation record should be ended using the "end" attribute set to "true".
    Remark: a situation record can be invalidated by use of the validityEndTime : refer to the notes on this attribute.
- in FilterExitManagement class, 2 attributes are defined:
  - **filterEnd**: this attribute, set to true, indicates that the filter, for which a previous record version has been published, has become inactive,
  - **filterOutOfRange**: This attribute is set to true when a previous version of this record has been published and now, for this new record version, the record goes out of the filter (example: the traffic jam length was 6 km and was sent, the filter threshold is currently set at 5 km, and the new version is 3 km => the value « 3 km » must be published with filterOutOfRange=true).

4.10. Elaborated data publication

4.10.1. Objectives and utilisation mechanisms

This publication is used to give information that is in some way elaborated or derived. Typically this is data which has been elaborated by a traffic centre from input data held in its database.

4.10.2. Content

This publication contains one or more elaborated data sets.

With this publication, attributes can be given in the InformationHeader class which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

A set of default values for “forecast”, “period” and “time” can be provided to clients which are applicable throughout the publication if no others are provided at the detailed level. This helps in minimising the size of publications where large numbers of values are sent which share common parameters.

Also to assist in minimising the volume of required values to be sent for traffic status covering wide expanses of a road network, ReferenceSettings can be provided to a client. ReferenceSettings comprise a reference to a predefined set of non-ordered locations (see Predefined Locations publication) and a default value that can be assumed by a client for the traffic status at all these locations if none is received. Thus a supplier only needs to send for those locations the actual values of traffic status which deviate from this default value.

Source information can be given (country, identifier, name, type, reliable) for each elaborated data.

A **validity** can also be given (as for situation records) composed of:

- An validity status which defines if the record has an “active” status or this status is defined according to periods of time (one can consider the third possible value “suspended” is not applicable for this kind of data element).
- One mandatory global first start time “overallStartTime”
- One optional global real end time “overallEndTime”
- Several periods, each having a start date and an end date, for which the elaborated data is active/valid, detailed with hours/days/weeks/months
- Exception periods, each having a start date and an end date, for which the elaborated data is not active/valid, detailed with hours/days/weeks/months.

The optional "overrunning" is not applicable for this kind of data element.

It can be indicated if the elaborated data is a forecast.

In case of fault when elaborating this data, it is possible to specify the type of fault (intermittent data values, no data available, spurious unreliable data values, unknown fault or other).

Each elaborated data item comprises a **Basic Data** value.

Basic data value is a general abstract class which has the following attributes

- Time precision, precision to which time of calculation or measurement is given
- Period (in seconds)
- Time, the time when basic data value was measured or elaborated.

Each Basic Data value can have a pertinent location. A pertinent location is one **group of locations** (refer “location” paragraph). Generally, this group of locations is identified by reference; refer the following paragraph concerning predefined locations.

Basic Data values are of one of the following types:

- o Travel Time data
- o Traffic Status
- o Traffic data
- o Weather data

The following paragraphs describe each of these basic data types:

Travel Time data. It has the following specific attributes:

- Travel time trend type: decreasing, increasing, stable
- Travel time type: best, estimated, instantaneous, reconstituted
- Vehicle type for this travel time.

It may be also associated with

- Travel time (in seconds)
- Free flow travel time
- Normally expected travel time,

as well as

- Free flow speed

Traffic Status. It has the following specific attribute:

- Traffic trend type: traffic building up, traffic easing, traffic stable, unknown.

and can be associated with

- Traffic status value: Impossible, congested, heavy, free flow, unknown if it differs from the reference value given for the publication

Traffic data value (usually sent as Measured Data, but may also be sent as Elaborated Data). It is structured as follows:

Traffic values can be classified on the basis of Vehicle characteristics.

There are 5 kinds of traffic values:

- Traffic headway
- Traffic flow
- Traffic speed
- Traffic concentration
- Individual vehicle measurements (not applicable for elaborated data).

Weather data value (usually sent as Measured Data, but may also be sent as Elaborated Data). It is structured as follows:

There are 7 types of weather values:

- Precipitation information
- Wind information
- Temperature information
- Pollution information
- Road surface condition information
- Visibility information.
- Humidity information

However the following point is of note for its use in the Elaborated Data publication:

- Traffic values and weather values are normally provided as Measured data, but may be provided as Elaborated data if appropriate (e.g. because values are provided by calculation).

4.11. Predefined locations publication

4.11.1. Objectives and utilisation mechanisms

This publication is used to predefine (sets of) locations which can then be referred to in any of the other publications by means of a single versioned reference attribute. This allows great simplification of those publications where repeating static or quasi-static locations are used, such as in Elaborated Data publications and Traffic View publications.

A set of locations defined as a container may contain one location or can be characterised as an Itinerary (ordered set of locations) or as a non-ordered group of locations. Predefined itineraries as well as a group of non-ordered locations contain several predefined locations. Each predefined location may be of any type, i.e. point, linear or area. Obviously an itinerary cannot include area-typed locations. Each set of predefined locations and each individual predefined location have its own unique versioned identifier and can be referenced in other publications.

The main utilisation is when publications are sent periodically and always concern the same locations, which are mostly itineraries or elementary linear locations (for instance, elaborated data and traffic views publications).

The utilisation mechanism is as follows:

- Step 1: the DATEX II client gets a predefined location publication with the locations identifiers and definitions,
- Step 2: when the DATEX II Supplier delivers publications to this client, it only uses the predefined location identifiers and the client is able to look up the identifier and obtain the real location details.

The advantage of this mechanism is to reduce the publication size.

When a predefined location set is modified, the DATEX II Supplier has to inform the concerned DATEX II Clients.

4.11.2. Content

With this publication, attributes can be given in the InformationHeader which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

The **predefined location container** (abstract) shall be derived into a predefined itinerary, a predefined group of non-ordered locations or simply a predefined location. All three have a **unique versioned identifier** and may have a **name.**

Itinerary and group of non-ordered locations are made up of **predefined locations**; each location has its own unique versioned identifier and may have its own name.

The definition of each predefined location uses the same location definition as the other publications (refer paragraph: “Location”). When predefining these locations, usage of other references is not allowed when it infers circular references.

4.12. Traffic View publication

4.12.1. Presentation

A traffic view is a snapshot of what happens on a group of non-ordered locations at a given time.

Before reading this paragraph, read paragraph 3.3: Publication of basic elements, where this publication is presented.

4.12.2. Content

With this publication, attributes can be given in the InformationHeader which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

A traffic view publication contains one or more **traffic views**.

Each traffic view has a **time** and refers to a **predefined location set**.

Each traffic view is organised in successive **linear traffic views**.

Each linear traffic view has a unique identifier and refers to a **linear predefined location**

Each linear traffic view can contain one or more **traffic view records**.

Each traffic view record has a sequence number and can be made of:

- A traffic element (refer to the situation publication section),
- An operator action (refer to the situation publication section),
- Elaborated data (refer to the elaborated data publication section),
- A reference to an Internet resource (e.g. camera image, video flow, RSS flow…).

4.13. Measured data publication

This publication contains measurements from one or more sites.

With this publication, attributes can be given in the InformationHeader which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

The measured data publication must contain a reference to the measurement site table used.

Each set of site measurement values must always contain a reference to the measurement site and a measurementTimeDefault which is the default time for each measured value if no other time is specified for an individual measurement.

Each set of site measurements has an ordered (indexed) list of measured values, where the order (index) corresponds with the measurement definition order (index) in the Measured Site table. So there is no need for any further identification for each measured value. The values that are measured are referenced by the index in the ordered list of values for that site.

A **measured value** can identify the type of measurement equipment used.

A measured value can also have a Location characteristics override. This is used to override the predefined lanes attribute and to indicate a reversal of flow compared with that was defined in the Measurement Site table.

In case of fault when measuring this data, it is possible to specify the type of fault (intermittent data values, no data available, spurious unreliable data values, unknown fault or other).

Each measured value comprises a **Basic data** value which may be one of the following types:

- Travel Time data
- Traffic Status
- Traffic data
- Weather data

Basic data value is a general abstract class which has already been described in the Elaborated Data section above. However the following points are of note for its use in the Measured Data publication:

- The point at which the measurement equipment is located (i.e. the site) is defined in the appropriate entry in the Measurement Site table. However, if the measurement relates to a section of road or a group of locations (e.g. ANPR which determines measurements over a stretch of road) this can also be provided in the “pertinentLocation” which is an aggregation associated with the **BasicData**.
- **TrafficStatus** and **TravelTimeData** are normally provided as Elaborated data, but may be provided as Measured data if appropriate (e.g. because values are provided by an outstation).

4.14. Measurement site table publication

4.14.1. Objectives and utilisation mechanisms

The Measurement Site Table publication is used to provide information relating to predefined measurement sites and the measurements that can be made at those sites. The sites and the details of their measurement characteristics are seen as static or quasi static (i.e. they seldom change).

The measurement site details and their measurement characteristics are referenced by Measured Data publications which enable these publications to be efficient when large volumes of measured data values need to be conveyed.

4.14.2. Content

This publication contains one or more measurement site tables.

With this publication, attributes can be given in the InformationHeader which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

The **Measurement Site table** has a **unique versioned identifier**.

The Measurement Site table has a list of **Measurement site records**, each corresponding to a single measurement site.

Every Measurement site record has a **unique versioned identifier** and the following attributes:

- Computation method,
- Measurement equipment reference,
- Measurement equipment type used,
- Measurement site name,
- Measurement site number of lanes,
- Measurement site identification,
- Measurement side,
- Measurement site record version and time.

The Measurement site record always has a single **Group Of Locations.**

Every site record must also have an ordered list of **Measurement specific characteristics**. This ordered (indexed) list of characteristics relates to the types of measurements that can be made at that measurement site. The ordering is significant because this order (indexing) is used in the Measured Data publications to distinguish between the different measurements received from the referenced site.

Measurement specific characteristics contains information about:

- Accuracy
- Period
- Smoothing factor
- Specific lane
- Specific measurement value type.
- Specific vehicle characteristics

These values are seen as static for the particular measurement at the site and do not have to be repeated for each measured value in the Measured data publication, unless they need to be overridden in particular cases (see Measured data publication section).

4.15. VMS publication

4.15.1. Objectives and utilisation mechanisms

This publication is used to give information on the current status and settings of one or more variable message signs (VMS) units, each unit controlling one or more individual variable message signs.

4.15.2. Content

This publication contains one or more VMS units.

With this publication, attributes can be given in the InformationHeader class which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

Each VMS unit must always contain a reference to a versioned VMS Unit table and a reference to a versioned VMS unit record in this VMS Unit table which defines the characteristics of the VMS unit. A VMS unit may control one or more variable message signs on a single gantry or on different gantries.

The VMS class provides the current status and settings of the VMS and the currently displayed information. Where a VMS is displaying a sequence or alternating set of messages these are ordered according to the messageIndex qualifier. Each VMS associated with a VMS unit is referenced by its vmsIndex. The vmsIndex also provides a reference to the specific VmsCharacteristics (defined in the VmsUnitTablePublication) that are relevant for the VMS which is controlled by the referenced VMS unit.

To assist in minimising the volume of required values to be sent for VMS messages covering wide expanses of a road network, characteristics of VMS and VMS units can be provided to a client using the VmsTablePublication.

In case of fault of VMS site or of VMS, it is possible to specify the type of fault (communication failure, power failure, out of service, incorrect message or pictogram display, unable to clear down, unknown fault or other).

A **VMS** is composed of:

- A general item if the VMS is working and about message sequencing interval in seconds, if applicable.
- The **Text Display Area Settings** (text lanterns, text luminance level, and luminance override).
- An ordered set of **Pictogram Display Area Settings**, one for each display area (picto lantern, picto luminance, …). Each pictogram display area on a VMS is referenced by its "pictogramDisplayAreaIndex" and details of its position relative to the text or its absolute position may be provided in **VmsPictogramDisplayCharacteristics**.
- An ordered set of **VmsMessages**. The messageIndex provides the ordering of a sequence of displayable messages. The VMS is displaying each message in turn according to its messageindex "order". The sequence of messages displayed is continuously cycling.

In case a VMS is not defined previously in a VmsTablePublication, its static characteristics must be provided about:

- Its location in term of position (vmsLocation – practically defined with a point)) and in term of managed logical location i.e. the road section (defined as a group of locations) where the VMS is applicable.
- Its physical characteristics with VMS text display characteristics and VMS pictogram display characteristics. As a VMS may have several pictogram areas, every one is referenced by its pictogramDisplayAreaIndex. Every area may be completed by **VMS supplementary panel characteristics**.

These latter classes are defined more in details in the subsequent **VMS table publication**.

Each elaborated data item comprises a **Basic Data** value.

Every **VMS message** comprises several elements. The adopted modelling takes into account different types of messages, with or without displayed text of several lines (in sequence), with or without pictogram display, each pictogram can have a **VMS supplementary panel**. The general information includes:

- VMS message information type (situation warning, travel time, traffic management, date and time, temperature, campaign message, instruction, future information).
- Time, the time when message was lst set
- Coded reason for setting (situation, campaign, operator created, traffic management, travel time or default) and the associated management or diversion plan.
- The authority, organisation or system which requested the setting of the message and the one which set it.
- References towards the related **situations** and **situation records** for which the message is related.
- Distance of the VMS from the location of the related **situation record**/element

A VMS message includes:

- One or several text pages. Where multiple pages of text are defined each is referenced by its "pageNumber" and is displayed sequentially according to the "pageNumber" order in the same display area.
- Each **VMS text** page comprises one or several lines of text. Where multiple lines of text are defined each is referenced by its "lineIndex" qualifier and is ordered accordingly. **The VMS text line** is also defined by the text itself, its colour, the used language, if it is flashing or not.
- One or several **VMS pictogram display areas**. Each pictogram display area on a VMS is referenced by its "pictogramDisplayAreaIndex" and details of its position relative to the text or its absolute position may be provided in VmsPictogramDisplayCharacteristics.
- Each **VMS pictogram display area** may display one pictogram or a sequence of pictograms. The order of sequencing of the pictograms is defined by the "pictogramSequenceIndex" values.
- Each **VMS pictogram** can be defined by one or several pictogram description, a code, the value of distance,, length, weight or width to be displayed on the pictogram, if the

pictogram is in inverse colour or if there is presence or red triangle or not, if it is compliant with the Vienna convention on road signs and signals, …

- A pictogram can be supplemented by one **VMS supplementary panel** which consists in one **VMS text line** (see above for description)

4.16. VMS table publication

4.16.1. Objectives and utilisation mechanisms

The VMS Table publication is a publication containing one or more VMS Unit Tables each comprising a set of records which hold details of VMS units. The VMS sites and the details of their characteristics are seen as static or quasi static (i.e. they seldom change).

The VMS tables and their VMS unit characteristics are referenced by VMS publications which enable these publications to be efficient when large volumes of displayed messages need to be conveyed.

4.16.2. Content

This publication contains one or more VMS unit tables.

With this publication, attributes can be given in the InformationHeader which provide clients with information management details (area of interest, confidentiality, information status, urgency) relating to this publication.

The **VMS unit table** has a **unique versioned identifier**.

The VMS unit table has a list of **VMS unit records**, each corresponding to a single VMS unit (entity managing one or several physical VMSs).

Every VMS unit record has a **unique versioned identifier** and the following attributes:

- Number of VMS,
- VMS unit identifier,
- VMS unit electronic address,
- VMS unit IP address,

Every VMS unit record must also have an ordered list of **VMS records** containing the physical characteristics and the location of each VMS. This ordered (indexed) list of these characteristics relates to different display areas on the VMS. The ordering is significant because this order (indexing) is used in the VMS publications to distinguish between the different display areas.

The VMS record can have until two **Groups Of Locations**. The first one gives the physical VMS location in term of position (vmsLocation – practically defined with a point)) and the second one gives its managed logical location i.e. the road section (defined as a group of locations – practically defined with one or several linears) where the VMS is applicable.

The VMS record also contains global information about:

- VMS type (colour graphic, monochrome graphic, continuous sign, matrix sign, other)
- Number of pictogram display areas
- VMS description
- VMS display height and width, its height above roadway
- VMS physical mounting (gantry, central reservation, roadside and roadside cantilever, overhead bridge, tunnel entrance, trailer and vehicle)
- VMS owner
- VMS type code.
- If its display areas are dynamically configurable.

It can also contain information for the VMS text display characteristics:

- Maximum font height, width and spacing
- Maximum numbers of characters, rows and sequential pages
- Maximum text luminance level
- Minimum font height, width and spacing

- Text display height and width
- Text lanterns presence
- Text page sequencing capability
- Number of text pixels across and down
- Text position absolute (on left, on right, at top, at bottom) and coordinates (X & Y) in metres
- Legend code list identifier

As well as for each VMS pictogram display characteristics through its "pictogramDisplayAreaIndex", details of its position relative to the text or its absolute position may be provided like:

- Maximum number of sequential pictograms
- Pictogram display height and width
- Pictogram number of colours
- Maximum pictogram luminance level
- Pictogram lanterns presence
- Pictogram sequencing capability
- Number of pictogram pixels across and down
- Pictogram position absolute (on left, on right, at top, at bottom), relative to text (above, below, to the left, to the right) and coordinates (X & Y) in metres
- Pictogram code list identifier

These values are seen as static for the particular VMS and do not have to be repeated for each displayed message in the VMS publication, unless they need to be overridden in particular cases (see VMS publication section).

4.17. Group of Locations

4.17.1. Important features

With DATEX II:

- information can be located with one group of locations where each location must be physically different,
- a group of locations can be an itinerary (i.e. an ordered group of locations), a group of non-ordered locations or a single location.
- each individual location can be defined with several location referencing systems (ALERT-C, linear referencing, TPEG-Loc, …): for interoperability, the Supplier and its Clients MUST use at least one common location referencing system which is a result of an interchange agreement (refer paragraph to Interchange_agreement),
- the groups of locations and each location can be expressed by references defined in the Predefined Locations Publication,
- A location can correspond to a road network element (point or linear) or to an area,
- An itinerary can relate to one or several destinations, each being a point or an area, It is also true for a road network location.
- When the location is on the network, supplementary positional description information can be given (e.g. characteristics at the carriageway and/or at the lane level, …)

More information is provided about the different location referencing systems in Annex 2.

4.17.2. Point location

Several location referencing systems can be used. At least one must be present:

- Point by coordinates (ETRS89[4]): latitude, longitude plus optional bearing

[4] Coordinates are defined according to the European Terrestrial Reference System 1989 (ETRS89) which was coincident with ITRS in 1989. This is the European implementation of ITRS, but unlike ITRS and WGS84 it is centred on Europe. ETRS89 is the EU-recommended frame of reference for geodata in Europe.

- Point along linear element: linear referencing system based on referents, distances to referents all of them being along a linear element (road element). Details (road name, direction, height, …) may also be added.

  The linear referencing system is presented in a separate paragraph.

- ALERT-C Point: country code, table number and version, plus:
  - Mandatory code and ALERT-C direction and optional name (Method 2);

  or

  - Same elements plus offset distance (Method 4)

  The ALERT-C location referencing system is presented in a separate paragraph.

- TPEG Point Location

  A TPEG point location has direction (e.g. northbound, eastbound, clockwise …) and is either specified as:

  - a simple point on the road network either at a junction or not,

  or

  - a framed point on the road network between two other points (i.e. framing points) on the road network.

  Each point (simple point, framed point or framing point) is defined by its latitude and longitude (ETRS89) which may be at a junction or between junctions. Each is further elaborated by one or more textual descriptors of prescribed type (i.e. names like junction names, point names, railways station name…).

4.17.3. Linear location

Several linear location referencing systems can be used. At least one must be present:

- Linear within linear element location: besides details common with the point along linear element, it includes:
  - A linearly referenced point identifying the “from” point and another linearly referenced point identifying the “to” point, where the “to” point is downstream from the “from” point according to the direction provided by the underlying linear road element

  The linear location referencing system is presented in a separate paragraph.

- ALERT-C linear: country code, table number and version, plus:
  - Mandatory linear location code and ALERT-C direction and optional name (method linear by code)

  or

  - Mandatory primary code, secondary code and ALERT-C direction and optional names (Method 2), plus offset distances (Method 4)

  The ALERT-C location referencing system is presented in a separate paragraph.

- TPEG Linear Location
  - A TPEG point location identifying the “from” point and another TPEG point location identifying the “to” point, where the “to” point is downstream from the “from” point according to the direction of traffic flow.
  - A direction (e.g. northbound, eastbound, clockwise …)
  - A fixed location type of “segment”

4.17.4. Area location

Several location referencing systems can be used. At least one must be present:

- ALERT-C area: country code, table number and version, plus:
  - Mandatory area location code and optional name

  The ALERT-C location referencing system is presented in a separate paragraph.

- TPEG Area Location

A TPEG area has a type of either “largeArea” or “other” and optional height information and height type (e.g. above sea level…). It is further specified either as:

- A “named only” area (i.e. by a name and a descriptor type, e.g. county name, town name…)

or

- A geometric area comprising a centre point (defined by ETRS89 latitude and longitude), a radius and an optional “named area”.

4.17.5. ALERT-C location referencing system

This section describes the ALERT-C referencing system and gives details of the contents of ALERT-C tables. Since this version of the DATEX II specification does not include the exchange of ALERT-C Location tables this is provided as general support information.

Reminder: An ALERT-C localisation is a tabular address defining a locating element belonging to this table.

Only the reference is transmitted between the Supplier and the Client. It is up to the client to look it up in the table for the descriptive elements corresponding to the referenced location using the ALERT-C code. To have a unique ALERT-C coding at the European level, the reference must comprise three elements (triplet):

- Country Code,
- Table number,
- Location number

Two distinct locations at the European level cannot have the same triplet.

Caution: currently certain codes for country table references were not allotted; they are thus invalid.

It is the European standard EN ISO 14819-3 that defines the rules of constitution of the ALERT-C location tables.

The following table gives indications on the contents of the various common attributes used by the classes related to the ALERT-C location:

Code	Definition	Possible values
alertCLocation CountryCode	Country code defined by RDS (IEC 62106). It uses only 15 values (1 hexadecimal digit: 1 to 9 and A to F). It doesn’t identify a specific country because values are shared by several countries.	F (France) 1 (Germany) E (Spain) 5 (Italy) etc…
alertCLocation TableNumber	Number allowed for a country CountryCode + TableNumber fully identifies the table	17 to 32 (France) 17 to 24 (Spain) 1 to 16 (Italy) etc …
alertCLocation TableVersion	Version number of the ALERT-C location table in a country.	
alertCDirection Coded	It is the concerned direction. Positive direction corresponds to the direction used when following positive linkage in the table	both negative positive unknown
alertCDirection Named	String for the direction name	
alertCDirection Sense	Indicates for circular routes (i.e. valid only for ring roads) the sense in which navigation should be made from the primary location to the secondary location, to avoid ambiguity. The value TRUE indicates the positive RDS direction, i.e. direction of coding of road.	true (ALERT-C positive direction) false (ALERT-C negative direction)
alertCLocation Name	Location name (redundant because already in the table)	
offsetDistance	The non-negative offset distance from the ALERT-C referenced point to the actual point.	
specificLocation	Unique code within the ALERT-C location table which identifies the specific point, linear or area location.	Number between 1 and 63487

AREA

In an ALERT-C location table an area location includes a name and its type can be found starting from the contents of the location table.

It can be used to locate certain surface phenomena like weather phenomena. It can also be used to provide an indication of the destination (when it corresponds to a city for example).

POINT

A point will be used to locate certain phenomena where dimension length is negligible such as for example accidents, incidents (broken down vehicles), obstacles, …. DATEX II proposes two methods of localization for the points, called method 2 and method 4 (the previous DATEX method 1 and method 3 have been abandoned):

- In method 2, the point is defined by the tabular address of the ALERT-C point;
- In method 4, the point is defined by the tabular address of the ALERT-C point, supplemented by the distance between this ALERT-C point and the point to be located.

In an ALERT-C location table an ALERT-C point includes a type and a point name as well as the name of intersecting roads at the point. Moreover, a reference towards the road (or the segment of road which carries it) is systematically added to the point and makes it possible to attach the name/number of road to this point. Consequently all points are attached to a road (and only one).

LINEAR

Linear will be used to locate certain phenomena or operator actions where dimension length is relevant, such as for abnormal traffic flow (queues, …), the bad condition of roads (e.g. road surface), …

DATEX II proposes three methods of localization for linear:

- In method 2, the linear one is defined by two points (primary location and secondary location), each point being defined only by the tabular address of the corresponding ALERT-C point;
- In method 4, the linear one is defined by two points (primary location and secondary location), each point being defined by the tabular address of the corresponding ALERT-C point, supplemented by the distance between this ALERT-C point and the end of the linear location.
- In method "linear by code", the linear is defined only by the tabular address of the corresponding linear ALERT-C location (road, street or segment).

The primary location is the nearest defined <u>downstream</u> point (for methods 2 or 4).

In an ALERT-C location table an ALERT-C linear location includes a type, the names of the origin and end of linear section as well as the associated road number/name.

4.17.6. Linear referencing system of locations

This section describes the linear referencing system used by DATEX II. It accommodates all the actual linear referencing systems adopted by road operators in Europe. It can be used for systems based on a distance measured from the linear element origin: kilometre point or its US equivalent mile point or true mileage as well as for the UK system named “chainage”, hectometre-points, or link offset. It also accommodates systems where there exist intermediate points from which the distance is measured: systems based on kilometre-post, milepost, reference post as well as cross-streets or control sections. It is fully compliant with a draft CEN ISO standard on the same topic (EN ISO 19148) being adopted, with the two following restrictions:

- Distances are only expressed in metres according to the DATEX II principle on units, unlike the draft standard where it is permissible to express them with different units.
- The lateral/vertical offset features are not implemented

<u>Reminder</u>: This system is based on predefined elements that need to exchanged and agreed by the Supplier and the Client before being used in a DATEX II exchange

Only the reference is transmitted between the Supplier and the Client. It is up to the client to look up the descriptive elements corresponding to the referenced location using the provided elements.

Such a linear referencing system includes three elements:

- general information on the location itself:
  - administrative area including the location,

    - geographic direction: e.g.: northbound, westbound, clockwise, …
    - relative traffic flow direction of the considered location regarding the underlying referenced linear element direction: aligned, opposite, both, unknown
    - height grade of linear section: above grade, at-grade, below grade

- Expression of distance of the referenced point: this expression of distance can be absolute (expressed from the origin of the road element), relative to an upstream intermediate point named "**from referent",** or interpolative (expressed by a percentage along the linear element and calculated from the linear element origin).

  In case of relative method, the "**from referent**" may also be associated with a "**toward referent**" (downstream), both bracketing the considered point.

  In case the considered point is bracketed by two referents, the direction defined as going from the "from referent" to the "toward referent" overrides the general direction of the linear element (see below). The location direction is thus compared to this direction.

  A referent is typified: boundary, intersection, reference marker, landmark, road node.

  Finally, each referent may be associated with ETRS89 geographic coordinates (latitude, longitude).

- The linear element underlying the location. It includes several attributes:
    - road number,
    - road name, one of the two attributes shall be provided,
    - linear element nature (optional): road, road section, slip-road, other
    - reference model which identifies the road network reference model which segments the road network according to specific business rules. It may be e.g. a graph describing the considered road network.
    - reference model version: version of the referenced road network model

  This underlying linear element is defined either with an identifier string (e.g. looking up in a map database) or as an ordered sequence of points with one at each extremity. These points are defined as **referent**. In this latter case, the sequence order provides the reference linear element direction.

POINT

A point will be used to locate certain phenomena where dimension length is negligible such as for example accidents, incidents (broken down vehicles), obstacles, …. In case of a point referencing, the general information is provided by the class "**PointAlongLinearElement**".

LINEAR

Linear will be used to locate certain phenomena or operator actions where dimension length is relevant, such as for abnormal traffic flow (queues, …), the bad condition of roads (e.g. road surface), …

This linear is defined by its two extremities: "from" point and "to" point. Unlike some other referencing based on points like ALERT-C, the order between "from" point and "to" point is only defined downstream by considering the underlying linear element direction. If the impacted traffic direction is aligned, the attribute "directionRelativeOnLinearSection" is used. The general information is provided by the class "**LinearWithinLinearElement**".

AREA

This referencing system cannot be used for referencing areas.

4.18. Extension on Parking information publications

This version 2.1 also contains the different elements corresponding to an approved extension incorporating the following two publications "ParkingFacilityTablePublication" (for the static information associated with each parking facility) and "ParkingFacilityStatusPublication" (for the corresponding dynamic information). However, due to the on-going review in ESG5 as well as potentially in a future standardisation work item this extension is subject to change. Any investment decision that includes the use of this extension should consider the efforts that would be required to modify software based on this extension in order to comply with the evolution of the model.

Exchange specifications

5. Exchange specifications

5.1. Exchange reference documents

For DATEX II, one of the objectives was to clearly split the content data modelling and the data exchange specification.

Two documents deal with data exchange:

- A high level document, called here [**Exchange PIM**], which is independent from technological platforms and presents, definitions, subsystems, use cases, metadata, and operating modes.
- A detailed specification, called here [**Exchange PSM**], which is specific for one platform which has been chosen to be for DATEX II, "Web Services over http".

These documents which contain the details on the exchange process may not be of direct interest for stakeholders who only want to know the main exchange features and the main options they have to decide. This chapter provides a brief overview of what DATEX II exchanges comprise.

5.2. Exchange Overview

5.2.1. Logical view

An exchange system between a Supplier and its Client(s) is composed of 2 main subsystems:

- A **Publisher** subsystem, which makes data available and creates the payload publications,
- A **Delivery** subsystem, which adds exchange specific information and performs the physical delivery.

The DATEX II exchange specification only describes the delivery subsystem.

A **subscription** specifies the payload type to be exchanged. It can be defined by the Supplier or by the Client, online or offline. DATEX II allows users to have the liberty to develop subscriptions with the refinements they need.

There are 3 possible **operating modes** for data delivery:

1	Publisher Push on occurrence	Data delivery is initiated by the Publisher every time data is changed
2	Publisher Push periodic	Data delivery is initiated by the Publisher on a cycle time basis
3	Client Pull	Data delivery is initiated by the Client and data is returned as a response

Remarks:

- operating mode "Publisher Push on occurrence":
  - o in this mode, for situation publications (events, operator actions, …), situation and situation records lifecycles have to be managed (update, cancel, and end). This is described in paragraph 4.9.6: "Special case: Publisher Push on occurrence",

  The main advantage of this mode is the response time (which can be measured in seconds).
- operating modes "Publisher Push on occurrence and Publisher Push periodic":
  - o after a link failure between the Supplier and his Client(s), recovery mechanisms have to be built (refer to [Exchange PSM].

Depending on operating mode and publication type, different **update methods** are possible:

singleElementUpdate	If an atomic part of the data has been changed, this atomic part, and only this atomic part, will be exchanged (e.g. only a single situation record update in a multi element situation)
allElementUpdate	If an atomic part of the data has been changed, all data associated with the atomic part will be exchanged (e.g. all situation records in a multi element situation)
snapshot	A snapshot contains all information available, at a given time, for a subscription

Remarks:

With “Client Pull” operating mode:

- Only “snapshot” update method is used,
- For situations publication, there is no need to manage situation and situation records lifecycles because the exchanged situations are only the active ones at the time of the snapshot (i.e. in the last received exchange),
- After a link failure, no recovery mechanism is needed (the first Pull automatically gets the latest versions).

OPTIONS: depending on his needs, a stakeholder has to decide, at this logical level:

- The refinement of subscriptions, from the “exchange everything” to sophisticated filtering,
- The operating modes he wants to implement (as Supplier and/or as Client), the Pull operating mode being the more simple to develop,
- The update methods he wants to implement, if he uses Publisher Push operating modes

5.2.2. Technical view

At this technical level, we distinguish 2 kinds of systems:

- Pull systems,
- Push systems

Pull systems

- Supplier: it can use either:
  - Web services over http, with DATEX II WSDL description,
  - Only a http server.
- Client: it can use either:
  - Web services over http, with DATEX II WSDL description,
  - Basic http requests, either GET or POST.

Push systems

- Supplier: it uses Web services over http, with DATEX II WSDL description,
- Client: it uses Web services over http, with DATEX II WSDL description.

OPTIONS: depending on his needs, the stakeholder has to decide, at this technical level, for the operating modes he has chosen:

- for Pull supplier systems, the usage of Web Services technologies and tools or basic http technology,
- for Pull client systems, the usage of Web Services technologies and tools or basic http technology.

Remark: specialists do not agree on the solutions which have the lowest cost for a Pull system: Web Services technology or basic http technology, the choices depend essentially on general policy for software development.

Interchange agreement

6. Interchange agreement

6.1. Definition

The interchange agreement is a document that describes bilateral agreements between a Supplier and a Client, needed for the exchange of DATEX II information. The interchange agreement should deal with all the information that is needed for the data exchange to take place. The description assumes that the information exchange follows the basic principles of DATEX II. Specifically it is assumed that two parties are involved: a Supplier providing content and a Client receiving content.

6.2. General

Time period throughout which the overall agreement is valid

The agreement shall state the time period for which the agreement is valid. It should define the date of commencement and may indicate the date of its ending.

Rules for terminating the agreement before the expiry time of the agreement

The agreement should specify the conditions under which the contract can be terminated prior to an expiry date.

Company/authority names, contact addresses, telephone, fax and E-mail details

The two parties shall provide the company and the authorities´ names, address and the operational contact with all related communication data such as telephone, fax numbers and email addresses.

6.3. Access

IP Address of the Client

The agreement may mention the IP address of the client which is to be used if the supplier is to implement IP address discrimination.

IP Address of the Supplier

The agreement shall mention the IP addresses on which the supplier's services will be available. Each supplier service may have different applicable conditions associated with it.

Supplier user name

The agreement should stipulate the user name that will be adopted by the supplier.

Client user name

The agreement should stipulate the user name that will be adopted by the client.

Password for Supplier to connect to Client

If the parties decide to use a password, this should be specified. If other means of access control are used these should be specified.

Password for Client to connect to Supplier

If the parties decide to use a password, this should be specified. If other means of access control are used they should be specified.

6.4. Data exchange

Choice of network for exchange of data and configuration

All technical information needed for the exchange of data and configuration between the parties should be described (Example: VPN details, router parameters, ...).

Options

The choices made by the supplier and the client have to be compatible:

- DATEX II model version and any extension,
- Operating mode,
- Update method

Location referencing systems

The choices made by the supplier and the client have to be compatible.

When using ALERT-C tables, they have to be exchanged first between the Supplier and the Client and kept compatible when they are updated. When using linear referencing with referents, these latter also have to be exchanged between supplier and clients.

6.5. Rights and obligations

Description of the quality of the data provided by Supplier.

Description of actions by Supplier in connection with the Supplier Server being down:

- actions when Supplier Server is down for planned maintenance,
- actions when Supplier Server is down following a system failure,
- actions when Supplier server starts up.

Description of actions by Client in connection with the Client system being down:

- actions when Client system is down for planned maintenance,
- actions when Client system is down following a system failure,
- actions when Client system starts up.

Description of payment mechanism for the service, if any.

Statement on liability in case of transmission of incorrect information.

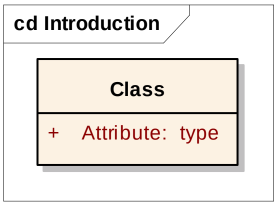
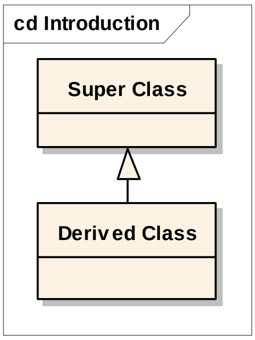
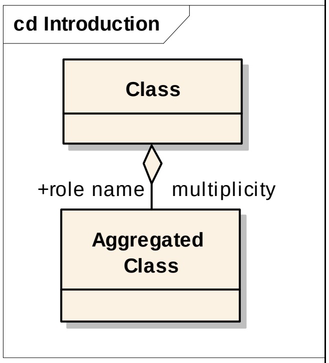
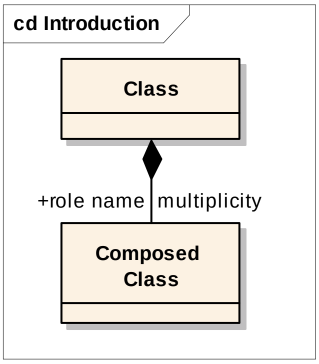
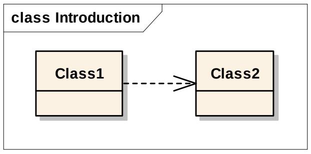
Rules for the further dissemination by the Client of information received using DATEXII.

Annexes

7. Annexes

7.1. Annex 1: UML basic elements

The following table shows a short description of UML 2.0 basic elements used by DATEX II. A full definition and specification can be found in UML reference documents.

Element type	Element	Description
Class		A class is a template for a given data element which can contain attributes. It is a rectangle divided into three compartments. The topmost compartment contains the name of the class. The middle compartment contains a list of attributes owned by that class and the bottom compartment contains a list of operations which is not shown here because operations are not used in this standard. In some diagrams, the bottom compartment of Attributes may be omitted for clarity reason. An attribute line has a specifier "+, # or –" for the visibility (not used in this standard) a name of the attribute and after a colon a data type and in squared brackets the multiplicity which is described in aggregation hereunder.
Specialisation		A Specialisation (i.e. Inheritance) defines a general class (super class) which properties are inherited from the derived class. In data structures that imply that the derived class has at least the same attributes as the super class and normally will define more attributes to it. Reason for using a inheritance in general is the capability of having different specialisations from one super class.
Aggregation	 cd Introduction Class +role name multiplicity Aggregated Class	The aggregation is a more explicit design element for describing attributes. It is a more strong association telling that the class on the side of the diamond "has" an instance of the aggregated class. The name of that instance is given on the left side of the connection and starts again with the "+" as specifier of visibility. On the right side the multiplicity of that instance is given as a range of the allowed count of occurrences. An aggregation does let open if the aggregated element has the same lifetime as the aggregating class. In data structures the aggregation can be a reference to another data structure or a embedded data element.
Composition	 cd Introduction Class +role name multiplicity Composed Class	The composition strengthens the type of aggregation in that way, that the lifetime of the composed element is the same as the composing class, i.e. the structure can be seen as a "composition". In data structures composition is normally seen as an embedded data element.
Dependency		The dependency is an unspecific type of relationship between two classes.

7.2. Use of ALERT-C location referencing system

See also EN ISO 14819-3:2004

7.2.1. Introduction

These rules are proposed to be used with method 4 (see § 4.17 above) that consists in providing ALERT-C point locations bracketing the element and distances from these points and the actual points. Method 2 is similar to method 4 but does not include offsets and thus the rules for offset do not apply.
The positive direction corresponds to the point (and segment when it exists)) order when going from the first point to the last point using positive offsets. It is named "AlertC chaining" in the pictures of the following examples. It fits the corresponding road direction defined as going from its first name to its second name.
For syntax, the user-defined values (between tags) are given within square brackets whereas the variable parameters (within a tag) are italicised. When a value between tags is in roman, it means the value comes from a predefined enumeration. When the possible values are all mentioned, the first follows the previous rule where as the others are in brackets, comma-separated.

7.2.2. Definitions for Method 4

- The primary point is the first ALERT-C point location after the head of the event in relation to the traffic direction.
- The primary point offset is the (positive) distance between the head of the event and the primary point.
- The secondary point is the first ALERT-C point location before the tail of the event in relation to the traffic direction.
- The secondary point offset is the (positive) distance between the tail of the event and the secondary point.
- The event direction, i.e. the corresponding traffic direction is given in related to the ALERT-C points chaining order: P: Positive, N: Negative, B: in both directions.
- Information concerning the primary point is mandatory for all events.
- Information concerning the secondary point is mandatory for linear events but optional for point events. (i.e. for space-limited locations)

NB1: Head and tail of an event are defined according to the usual congestion terminology.
NB2: An accident generally is on a point but may be of some extend and thus linear.

7.2.3. Syntax

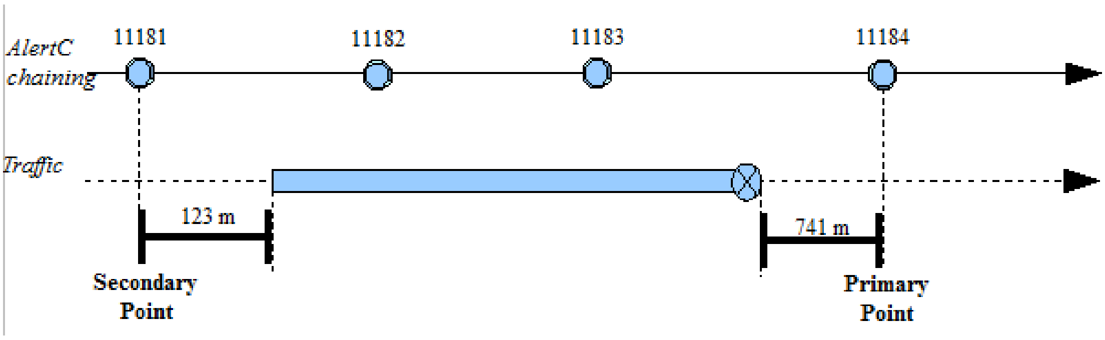
```
<alertCLinear xsi:type="AlertCMethod4Linear">
                              (or  <alertCLinear xsi:type="AlertCMethod4Point">
 <alertCLocationCountryCode>[F]</alertCLocationCountryCode>
 <alertCLocationTableNumber>[32]</alertCLocationTableNumber>
 <alertCLocationTableVersion>[alertC table version]</alertCLocationTableVersion>
    <alertCDirection>         (i.e. Event direction)
        <alertCDirectionCoded>both (positive, negative)</alertCDirectionCoded>
    </alertCDirection>
    <alertCMethod4PrimaryPointLocation>        (i.e. Primary point)
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">[Name of primary point]</value>
            </alertCLocationName>
            <specificLocation>[Primary point identifier]</specificLocation>
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>[Offset of primary point]</offsetDistance>
        </offsetDistance>
    </alertCMethod4PrimaryPointLocation>
    <alertCMethod4SecondaryPointLocation>          (i.e. Secondary point)
```

```
                <alertCLocation>
                    <alertCLocationName>
                        <value lang="fr">[Name of secondary point]</value>
                    </alertCLocationName>
                    <specificLocation>[Secondary point identifier]</specificLocation>
                </alertCLocation>
                <offsetDistance>
                    <offsetDistance>[Offset of secondary point]</offsetDistance>
                </offsetDistance>
            </alertCMethod4SecondaryPointLocation>
        </alertCLinear>
```

To simplify examples presentation, the following part of the syntax is not duplicated:

```
<alertCLocationCountryCode>[country code]</alertCLocationCountryCode>
<alertCLocationTableNumber>[table number]</alertCLocationTableNumber>
<alertCLocationTableVersion>[ table version]</alertCLocationTableVersion>
```

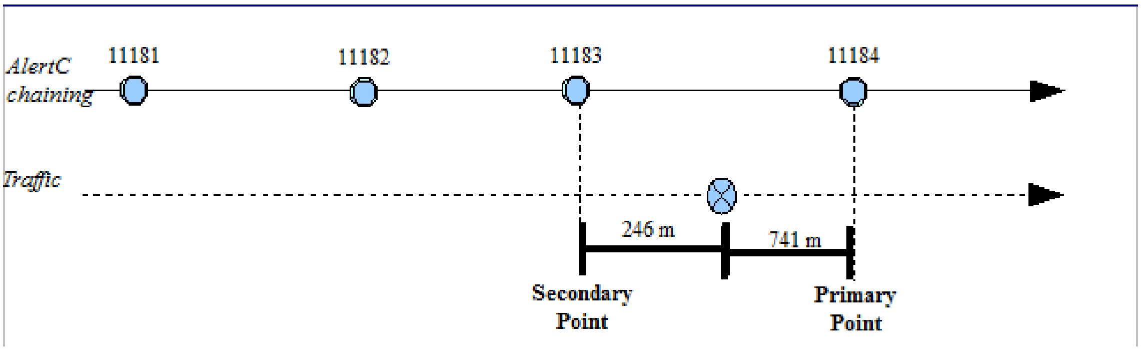
7.2.4. Example 1: Case of positive direction for a linear event



```
<alertCLinear xsi:type="AlertCMethod4Linear">
    <alertCDirection>
        <alertCDirectionCoded>positive</alertCDirectionCoded>
    </alertCDirection>
    <alertCMethod4PrimaryPointLocation>
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">Sémecourt</value>
            </alertCLocationName>
            <specificLocation>11184</specificLocation>
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>741</offsetDistance>
        </offsetDistance>
    </alertCMethod4PrimaryPointLocation>
    <alertCMethod4SecondaryPointLocation>
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">Aire de Metz Saint-Privat</value>
            </alertCLocationName>
```

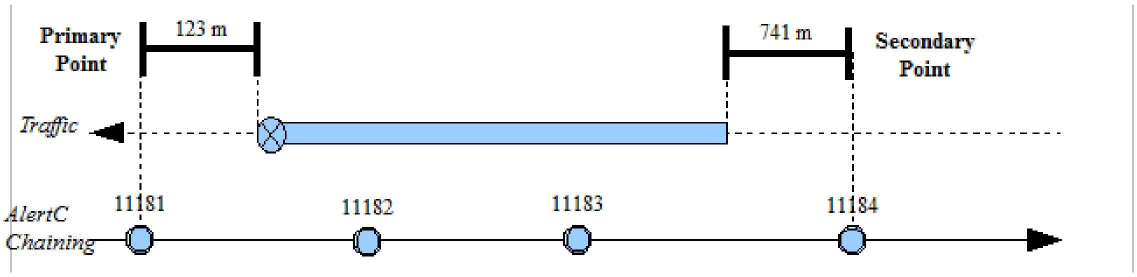
```
                <specificLocation>11181</specificLocation>
            </alertCLocation>
            <offsetDistance>
                <offsetDistance>123</offsetDistance>
            </offsetDistance>
        </alertCMethod4SecondaryPointLocation>
    </alertCLinear>
```

7.2.5. Example 2: Case of positive direction for a point event



```
<alertCLinear xsi:type="AlertCMethod4Point">
  <alertCDirection>
        <alertCDirectionCoded>positive</alertCDirectionCoded>
  </alertCDirection>
  <alertCMethod4PrimaryPointLocation>
        <alertCLocation>
              <alertCLocationName>
                    <value lang="fr">Sémecourt</value>
              </alertCLocationName>
              <specificLocation>11184</specificLocation>
        </alertCLocation>
        <offsetDistance>
              <offsetDistance>741</offsetDistance>
        </offsetDistance>
  </alertCMethod4PrimaryPointLocation>
</alertCLinear>
```

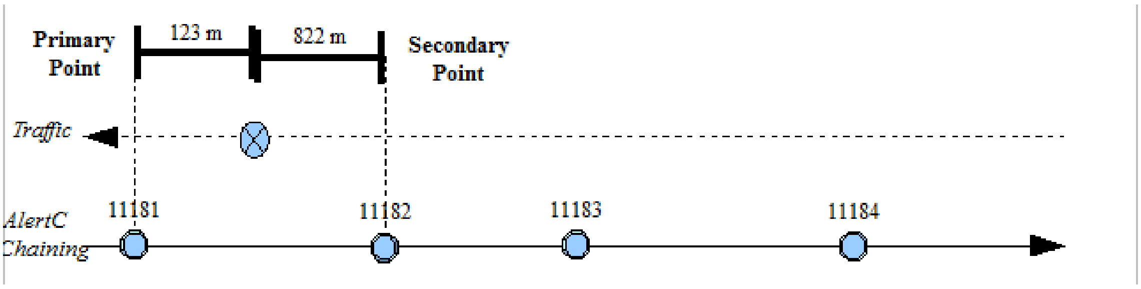
7.2.6. Example 3: Case of negative direction for a linear event



```
<alertCLinear xsi:type="AlertCMethod4Linear">
  <alertCDirection>
        <alertCDirectionCoded>negative</alertCDirectionCoded>
  </alertCDirection>
```

```
    <alertCMethod4PrimaryPointLocation>
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">Aire de Metz Saint-Privat</value>
            </alertCLocationName>
            <specificLocation>11181</specificLocation>
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>123</offsetDistance>
        </offsetDistance>
    </alertCMethod4PrimaryPointLocation>
    <alertCMethod4SecondaryPointLocation>
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">Sémecourt</value>
            </alertCLocationName>
            <specificLocation>11184</specificLocation>
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>741</offsetDistance>
        </offsetDistance>
    </alertCMethod4SecondaryPointLocation>
</alertCLinear>
```

7.2.7. Example 4: Case of negative direction for a point event

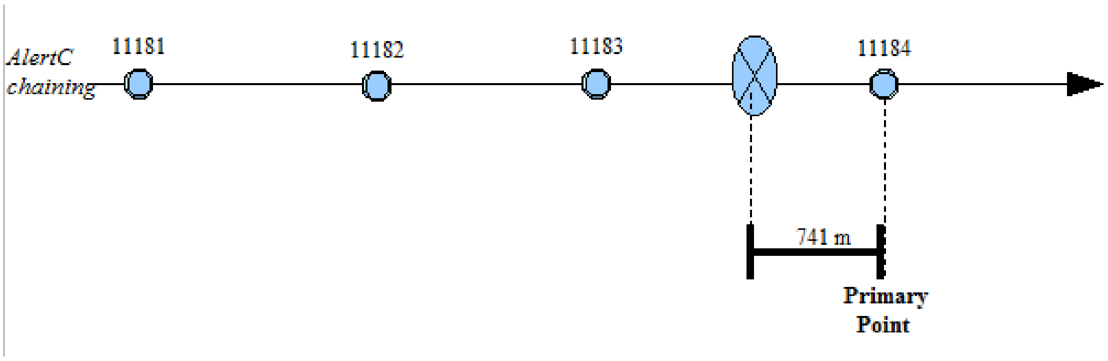


```
<alertCLinear xsi:type="AlertCMethod4Point">
    <alertCDirection>
        <alertCDirectionCoded>negative</alertCDirectionCoded>
    </alertCDirection>
    <alertCMethod4PrimaryPointLocation>
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">Aire de Metz Saint-Privat</value>
            </alertCLocationName>
            <specificLocation>11181</specificLocation>
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>123</offsetDistance>
        </offsetDistance>
    </alertCMethod4PrimaryPointLocation>
</alertCLinear>
```

7.2.8. Example 5: Case of both directions for a point event

In such case the traffic direction is not be taken into account.

But the Primary point should be always determined using the positive direction.



This event is located as follows:

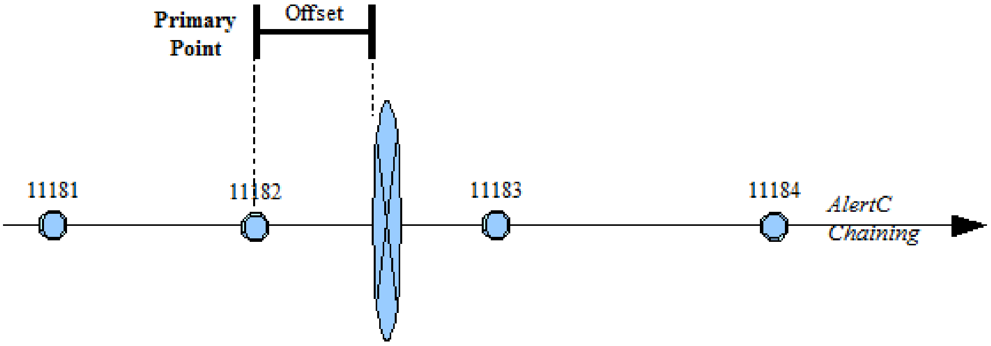
```
<alertCLinear xsi:type="AlertCMethod4Point">
    <alertCDirection>
            <alertCDirectionCoded>positive</alertCDirectionCoded>
    </alertCDirection>
    <alertCMethod4PrimaryPointLocation>
            <alertCLocation>
                    <alertCLocationName>
                            <value lang="fr">Sémecourt</value>
                    </alertCLocationName>
                    <specificLocation>11184</specificLocation>
            </alertCLocation>
            <offsetDistance>
                    <offsetDistance>741</offsetDistance>
            </offsetDistance>
    </alertCMethod4PrimaryPointLocation>
</alertCLinear>
```

Important remark

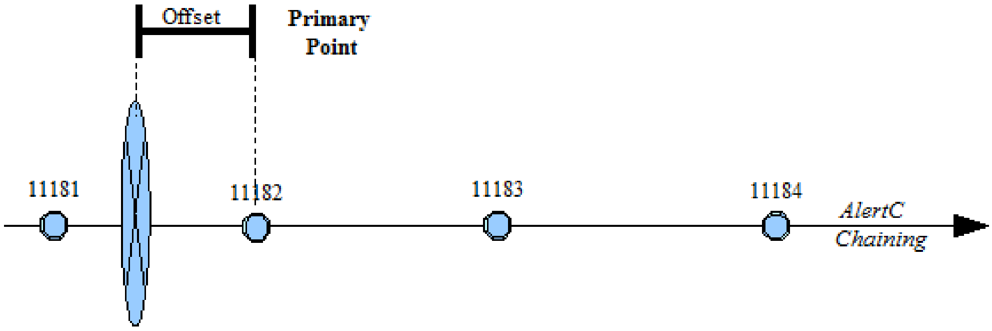
Note that in case a previous version of this event was coded as one-direction event, the primary point location would be the same for the positive direction .If the previous version was coded in the negative direction, the primary point location would be moved from 11183 (previous version) to 11184 (version in both direction).

If the primary point is not taken by reference to positive direction, there may be a misunderstanding by the client system as explained below:

1. The supplier application only generates the Primary point, based on the negative direction:



2. The client application interprets this message using the positive direction:

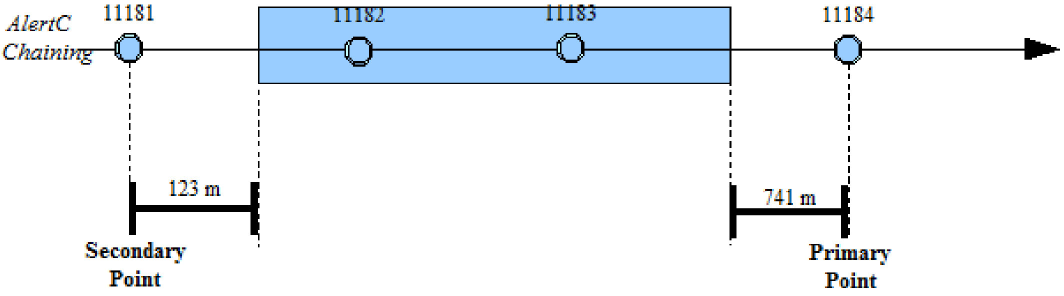


The event is thus shifted by twice the offset from its true position, often several kilometres.

7.2.9. Example 6: Case of both directions for a linear event

In such case the traffic direction is not taken into account.

However, the Primary point should be determined using the positive direction. This choice corresponds with the case of Point event, even if there is no risk of wrongly locating the event between supplier and client.



This event is located as follows:

```
<alertCLinear xsi:type="AlertCMethod4Linear">
    <alertCDirection>
            <alertCDirectionCoded>both</alertCDirectionCoded>
    </alertCDirection>
    <alertCMethod4PrimaryPointLocation>
            <alertCLocation>
                    <alertCLocationName>
                            <value lang="fr">Sémecourt</value>
                    </alertCLocationName>
                    <specificLocation>11184</specificLocation>
```

```
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>741</offsetDistance>
        </offsetDistance>
    </alertCMethod4PrimaryPointLocation>
    <alertCMethod4SecondaryPointLocation>
        <alertCLocation>
            <alertCLocationName>
                <value lang="fr">Aire de Metz Saint-Privat</value>
            </alertCLocationName>
            <specificLocation>11181</specificLocation>
        </alertCLocation>
        <offsetDistance>
            <offsetDistance>123</offsetDistance>
        </offsetDistance>
    </alertCMethod4SecondaryPointLocation>
</alertCLinear>
```

7.3. Use of linear location referencing system

7.3.1. Introduction

These rules are proposed to be used in case of linear referencing systems. They may be absolute, relative i.e. based on referents or interpolative (see section 4.16.6).

For syntax, the user-defined values (between tags) are given within square brackets whereas the variable parameters (within a tag) are italicised. When a value between tags is in roman, it means the value comes from a predefined enumeration. When the possible values are all mentioned, the first follows the previous rule where as the others are in brackets, comma-separated.

A linear reference is always attached to a linear element: it can be a predefined whole road or a continuous road section. It can also be defined as a link-node sequence with its nodes or reference posts. In this latter case a referent shall be given for both extremities.

The direction of an underlying linear element is defined by the referent (i.e. node) sequence according to the sequence order. Generally each referent has a one-to-one relationship with the consequent referent (or sometimes with the previous one). When using a link–node graph for underlying linear elements, the link direction (supposed to be oriented) is the reference direction.

Unlike the ALERT-C location referencing system, the traffic event does not have an influence on the way to define the corresponding location. Only one of the corresponding direction attributes of the location is dealt with.

Even if the corresponding attributes are not mandatory it is very wise to give either the road number if it exists or by default its name.

7.3.2. Definitions

- The “from” point of a linear is the first encountered point delineating a linear location according to the direction of the underlying linear element.
- The “to” point of a linear is the last encountered point delineating a linear location according to the direction of the underlying linear element.
- The distance is the curvilinear abscissa measured between the measure origin and the considered point. It is positive when it is made following the same direction as the one of the underlying element except for the interpolative method.
- For the interpolative method the distance is expressed as a percentage between the curvilinear abscissa from the origin and the total linear element length.

NB, an accident is generally a point event but may be of significant length in some cases and thus linear (example of a pile-up).

NB: according to the Datex II specifications, the event direction can be given either geographically ("directionBound") or by comparing the underlying element direction ("directionRelative").

All examples below use road as underlying linear element, its direction and its road number.

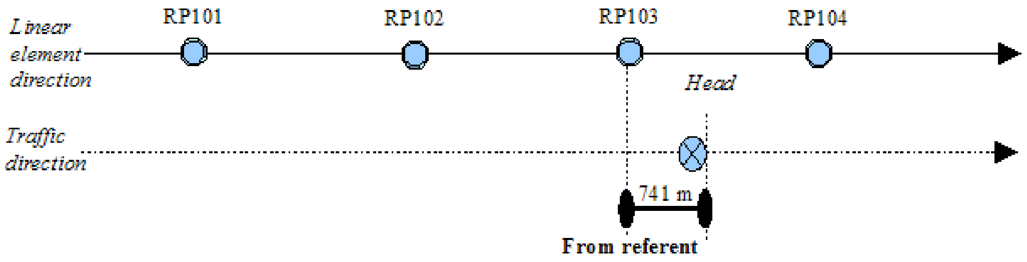
7.3.3. Syntax for a point location

```
<pointAlongLinearElement>
  <directionRelativeAtPoint> aligned (opposite, both)</directionRelativeAtPoint>
  <linearElement>
        <roadNumber>[Road number]</roadNumber>
        [Definition of this element either by a code or by a set a points – see below]
  </linearElement>
  <distanceAlongLinearElement xsi:type="DistanceFromLinearElementReferent">
        <distanceAlong>[Distance from referent]</distanceAlong>
        <fromReferent>
              <referentIdentifier>[Referent identifier]</referentIdentifier>
              <referentType>referenceMarker, (border, …)</referentType>
        </fromReferent>
  </distanceAlongLinearElement>
</pointAlongLinearElement>
```

7.3.4. Syntax for a linear location

```
<linearWithinLinearElement>
  <directionRelativeOnLinearSection>aligned (opposite, both)</directionRelativeOnLinearSection>
  <linearElement>
        <roadNumber>[Road number]</roadNumber>
        [Definition of this element either by a code or by a set a points]
  </linearElement>
  <fromPoint xsi:type="DistanceFromLinearElementReferent">
        <distanceAlong>[Distance from first referent]</distanceAlong>
        <fromReferent>
              <referentIdentifier>[First referent identifier]</referentIdentifier>
              <referentType>referenceMarker, (border,…)</referentType>
        </fromReferent>
  </fromPoint>
  <toPoint xsi:type="DistanceFromLinearElementReferent">
        <distanceAlong>[Distance from second referent]</distanceAlong>
        <fromReferent>
              <referentIdentifier>{Second referent identifier]</referentIdentifier>
              <referentType>referenceMarker, (border,…)</referentType>
        </fromReferent>
  </toPoint>
</linearWithinLinearElement>
```

7.3.5. Example 1: Case of aligned direction for a point event

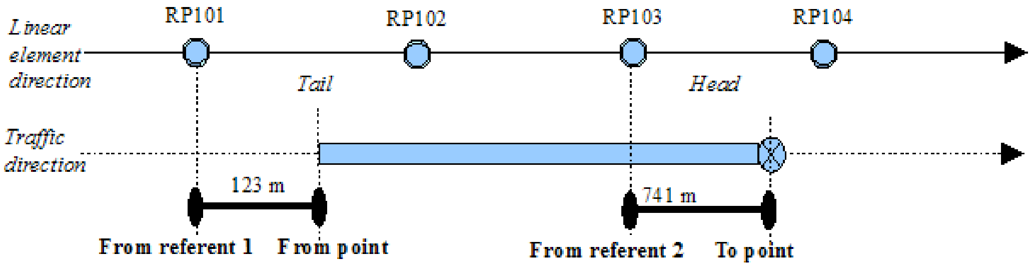


This event is located as follows:

- A point accident located using a ”from referent” RP103, a 741-metre abscissa and a aligned direction.

In case the event is in the opposite direction or in both directions the attribute “directionRelativeAtPoint” is modified accordingly. All the other elements / attributes are unmodified.

7.3.6. Example 2: Case of aligned direction for a linear event

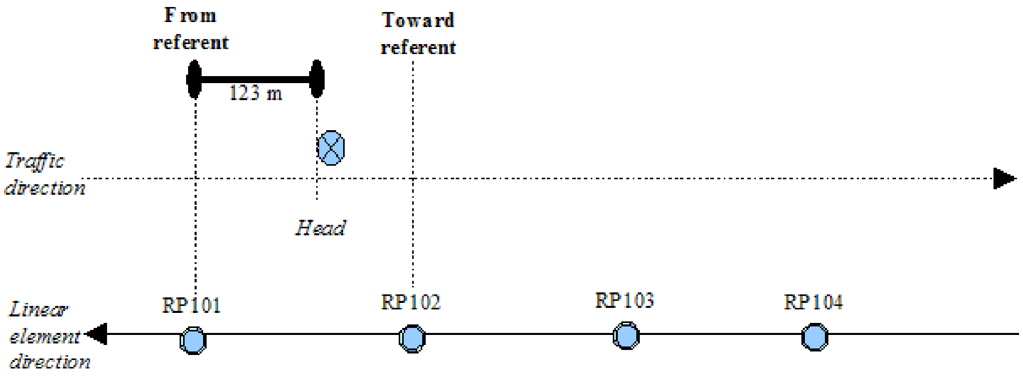


This event is located as follows:

- Congestion is located linearly with an aligned direction, a “from point” using the “from referent” RP101 and a 123-metre abscissa, a “to point” using the “from referent” RP103 and a 741-metre abscissa.

In case the event is in the opposite direction or in both directions the attribute “directionRelativeOnLinearSection” is modified accordingly. All the other elements / attributes are unmodified.

7.3.7. Example 3: Case of direction defined by two referents for a point event



This event is located as follows:

- Point accident with point located using a “from referent” RP101, a “toward referent” RP102, a direction going from RP101 to Rp102, a relative direction positive and a 123-metre abscissa measured from RP101.

7.3.8. Definition of the underlying linear element

As already stated above, the underlying linear element may be defined with a reference (code) to an external database which contains predefined linear elements e.g. links or GDF road elements.

The other possibility is to define “on the fly” this linear element as an ordered sequence of referents, the first one being placed at the start point of this element and the last one at the end point.

The corresponding syntax is the following (case of an intermediate point for this element):

```
<linearElement xsi:type="LinearElementByPoints">
   <roadNumber>[Road number]</roadNumber>
   <linearElementNature>road, roadSection, slipRoad, other</linearElementNature>
   <startPointOfLinearElement>
           <referentIdentifier>[First referent identifier]</referentIdentifier>
           <referentType>boundary, referenceMarker, …</referentType>
   </startPointOfLinearElement>
   <intermediatePointOnLinearElement index="1">
           <referent>
                   <referentIdentifier>[Intermediatet referent identifier]</referentIdentifier>
                   <referentType>boundary, referenceMarker, …</referentType>
           </referent>
   </intermediatePointOnLinearElement>
   <endPointOfLinearElement>
           <referentIdentifier>[Last referent identifier]</referentIdentifier>
           <referentType>boundary, referenceMarker, …</referentType>
   </endPointOfLinearElement>
</linearElement>
```

7.4. Use of TPEG location referencing system

7.4.1. Introduction

This part describes the minimum attribute set of TPEG location. This location type can be used extensively but as it is based on geographic map make sure that the client is able to understand all the attributes.
For syntax, the user-defined values (between tags) are given within square brackets whereas the variable parameters (within a tag) are italicised. When a value between tags is in roman, it means the value comes from a predefined enumeration. When the possible values are all mentioned, the first follows the previous rule where as the others are in brackets, comma-separated.

7.4.2. Definitions

- The head of the event is defined as the point where the generating factor of the event is situated. For example, in a traffic jam, the head is the point where traffic flows normally again after travelling through the congestion. The tail is situated at the point where the vehicle begins to slow down and enters the congestion.
- Unlike the ALERT-C location referencing system, the event is not defined between two referent points, but at a precise, independent position.
- In case of an isolated event, only the corresponding point is described with its direction.
- In case of a linear event, only the start and end points of the event are described. Direction is also given.
- TPEG referencing is based on a large number of classes, most of which are optional. The following chapter describes some implementation choices of these classes for a minimum use of TPEG.

As for the ALERT-C and linear referencing locations, a TPEG location in DATEX II must be contained within the tags:

```
<groupOfLocations>
</groupOfLocations>
```

NB: the following syntaxes and examples are based on non-junction points. However the principle can be adapted to cover the case of junction points.

7.4.3. Syntax for a point location

In case of a point event, the Datex II syntax is as follows:

```
<groupOfLocations>
  <locationContainedInGroup xsi:type="ns1:Point">
      <tpegPointLocation xsi:type="ns1:TPEGSimplePoint">
          <tpegDirection>[cardinal direction]</tpegDirection>
          <tpegSimplePointLocationType>nonLinkedPoint</tpegSimplePointLocationType>
          <point xsi:type="ns1:TPEGNonJunctionPoint">
              <pointCoordinates>
                  <latitude>[latitude]</latitude>
                  <longitude>[longitude]</longitude>
              </pointCoordinates>
              <name>
                  <descriptor>
                      <value lang="lang">[Name of town]</value>
                  </descriptor>
                  <tpegOtherPointDescriptorType>
                      townName
                  </tpegOtherPointDescriptorType>
              </name>
              <name>
                  <descriptor>
                      <value lang="lang">[Name of road]</value>
                  </descriptor>
                  <tpegOtherPointDescriptorType>
                      linkName
                  </tpegOtherPointDescriptorType>
              </name>
          </point>
      </tpegpointLocation>
  </locationContainedInGroup>
</groupOfLocations>
```

NB: The attribute "tpegSimplePointLocationType" may have one the following two values depending on the location type: "junctionPoint" (when located on an intersection) and "nonLinkedPoint" (otherwise). On a junction the value "junctionPoint" shall be used and either the junction name or the names of the intersecting roads (3 maximum) shall be given.

7.4.4. Syntax for a linear location

In case of a linear event, two points are described with the DATEX II syntax below. The point "to" corresponds with the head of the event. The point "from" corresponds to the tail of the event.

```
<groupOfLocations>
  <locationContainedInGroup xsi:type="ns1:Linear">
      <tpegLinearLocation>
          <tpegDirection>[cardinal direction]</tpegDirection>
          <tpegLocationType>segment</tpegLocationType>
          <to xsi:type="ns1:TPEGNonJunctionPoint">
              <pointCoordinates>
                  <latitude>[latitude at head]</latitude>
```

```
                    <longitude>[longitude at head]</longitude>
                </pointCoordinates>
                <name>
                    <descriptor>
                        <value lang="lang">[Name of town at head]</value>
                    </descriptor>
                    <tpegOtherPointDescriptorType>
                        townName
                    </tpegOtherPointDescriptorType>
                </name>
                <name>
                    <descriptor>
                        <value lang="fr">[Name of road at head]</value>
                    </descriptor>
                    <tpegOtherPointDescriptorType>
                        linkName
                    </tpegOtherPointDescriptorType>
                </name>
            </to>
            <from xsi:type="ns1:TPEGNonJunctionPoint">
                <pointCoordinates>
                    <latitude>[latitude at tail]</latitude>
                    <longitude>[longitude at tail]</longitude>
                </pointCoordinates>
                <name>
                    <descriptor>
                        <value lang="lang">[Name of town at tail]</value>
                    </descriptor>
                    <tpegOtherPointDescriptorType>
                        townName
                    </tpegOtherPointDescriptorType>
                </name>
                <name>
                    <descriptor>
                        <value lang="lang">[Name of road at tail]</value>
                    </descriptor>
                    <tpegOtherPointDescriptorType>
                        linkName
                    </tpegOtherPointDescriptorType>
                </name>
            </from>
        </tpegLinearLocation>
    </locationContainedInGroup>
</groupOfLocations>
```

7.4.5. Expression of latitude and longitude

These are expressed in decimal degrees in the European Terrestrial Reference System 1989 (ETRS89) which complies with ITRS, thus the values are considered as being equivalent.