ROSATTE

**RO**ad Safety **ATT**ributes exchange infrastructure in **E**urope

# *Specification of data exchange methods*

Version Number : 16 (final version)
Produced by : Triona / NAVTEQ
Due Date : 31-12-2008
Release Date : 31-08-2009

ROSATTE is co-financed by the European Commission – DG INFSO
Contract Number: 213467

| | |
|---|---|
| **Programme name** | 7th Framework Programme – Specific Programme Cooperation Theme 3 "Information and Communication Technologies" |
| **Grant agreement number:** | 213467 |
| **Project acronym:** | ROSATTE |
| **Project name:** | ROad Safety ATTributes exchange infrastructure in Europe |
| **Project start and end:** | January 2008 – June 2010 (30 months) |
| **EC Project Officer:** | Mrs. Elina Holmberg |
| | E-mail: Elina.HOLMBERG@ec.europa.eu |
| **Dissemination level[1]:** | PU |

| | |
|---|---|
| **Title of document:** | Specification of data exchange methods |
| **Work package:** | WP3 |
| **Author(s):** | <see Document control sheet> |
| **Project co-ordinator:** | Maxime Flament (ERTICO – ITS Europe) |
| | Tel: +32 2 400 07 35, fax: +32 2 400 07 01 |
| | E-mail: m.flament@mail.ertico.com |

**Abstract:** The document provides the specification of the ROSATTE data exchange framework, with the following core elements: definition and description of relevant safety attributes; location identification; definition and description of update types; message formatting and packaging; web-based data exchange services.

**Keyword list:** road safety feature, road safety attribute; location referencing; update type; data message format; data packaging format; exchange service

[1] This is either: Public, restricted to other programme participants, restricted to a group specified by the consortium, confidential

# Document control sheet

Authors: Lars Wikström, Triona (editor)
Michael Landwehr, PTV
Hendrik Bock, PTV
Ahmed Nasr, NAVTEQ
Kees Wevers, NAVTEQ
Per-Olof Svensk, Triona
Stephen T'Siobbel, Tele Atlas
Bert Boterbergh, Vlaamse Overheid
Loïc Blaive, SETRA
Hamish Keith, Traffic For London
Per Isaksson, SRA
Olof Bergman, SRA
Ulrich Haspel, OBB
Rainer Schützle, University of Stuttgart

Work area: WP3

Document title: Specification of data exchange methods

Document file name: RST-D31-Specification-of-data-exchange-methods-v16.doc/pdf

**Version history:**

| Version | Date | Main author | Summary of changes |
|---|---|---|---|
| 01 | 05-11-2008 | Kees Wevers | Creation of first outline |
| 02 | 25-11-2008 | Kees Wevers | Second draft version for discussion 25-11-08 |
| 03 | 04-12-2008 | Kees Wevers | Adapted structure of document |
| 04 | 15-12-2008 | Lars Wikström | Added proposals for content in chapters regarding:<br>- Content spec<br>- Data exchange spec<br>- Service spec<br>Moved sections safety attributes, location referencing, update types and metadata to a new background chapter according to proposal from P-O Svensk. |
| 05 | 29-01-2009 | Lars Wikström | Added concrete proposal for the content of chapters 6 and 7 |
| 06 | 03-02-2009 | PTV | Added 1st proposal for chapter 8 content |
| 07 | 18-02-2009 | Lars Wikström | - Added necessary definitions for speed limit<br>- Added definitions on allowable safety attribute property types for safety |

| | | | |
|---|---|---|---|
| | | | attribute types<br>- Added 1st proposal for AGORA-like data structure for location referencing<br>- Safety attribute -> safety feature according to TFL comment.<br>- Changed definitions for updates according to Brussels meeting to a safet feature centric approach – update primitive as attribute in safety attribute.<br>- Modifications to the feedback package according to SRA comments<br>- Added new chapter 8 from PTV |
| 08 | 20-02-2009 | Lars Wikström | Added dependency between safety features and conditions. |
| 09 | 03-03-2009 | Lars Wikström | Added input from WP members. |
| 10 | 07-03-2009 | Lars Wikström | Review version according to decided tasks at the Malmö meeting 2009-03-25 |
| 11 | 15-05-2009 | Lars Wikström | Updated according to decisions made during the Munich meeting 2009-05-06 |
| 12 | 15-05-2009 | Ahmed Nasr | Layout, format (minor modifications)<br>Version sent for Peer review |
| 13 | 30-06-2009 | Lars Wikström | Edited according to peer review comments resolution |
| 14 | 17-07-2009 | Ahmed Nasr | Update figure numbering and references, editing and formatting |
| 15 | 03-08-2009 | Ahmed Nasr | Update "Purpose of the document" to "Excutive summary", update figure 5 |
| 16 | 31-08-2009 | Ahmed Nasr | Move project description & Excutive summary before the Table of Contents, update the references section |

**Approval:**

| | Name | Date |
|---|---|---|
| Prepared | Kees Wevers, Ahmed Nasr (NAVTEQ) | 15-05-2009 |
| Reviewed | Peer reviewers | 31-08-2009 |
| Authorised | Maxime Flament (ERTICO) | 04-09-2009 |

**Circulation:**

| Recipient | Date of submission |
|---|---|
| ROSATTE Consortium | 03-09-2009 |
| EC | 05-10-2009 |

# Project Description

## *ROSATTE Contractual References*

ROSATTE is a STREP submitted for the call FP7-ICT-2007-1. The acronym stands *for ROad SAfety ATTributes exchange infrastructure in Europe*. The Grant Agreement number is 213467 and project duration is 30 months, effective from 01 January 2008 until 30 June 2010. The agreement is with the European Commission, DG INFSO.

The EC Project Officer is:

***Elina Holmberg***
EUROPEAN COMMISSION
DG INFSO - G04
Office: BU31 - 5/41
B - 1049 Brussels
Tel: +32 2 298 76 12
E-mail: Elina.HOLMBERG@ec.europa.eu

## *Project objectives and scope*

The ROSATTE project intends to develop the enabling infrastructure and supporting tools that will ensure Europe-wide access to safety road attributes with a focus on incremental updates. This infrastructure is intended to facilitate a continuous supply of such data at a high and steady level of quality from the parties that administer and control the attributes to third parties, and thereby to help maintain near-permanent correctness of such data for use in road safety applications. In addition, the infrastructure will serve administrative internal functions at data providers, which will in turn be beneficial for the system of safety road attributes as a whole. Improved and more extended availability of up-to-date safety road attributes is expected to result in improved and extended functionality of driving assistance systems, and thereby to contribute to more efficient road operations and increased traffic safety.

The flow of data that is addressed in ROSATTE may be seen as a data chain, which is depicted in the upper part of Figure 1. Public road administrations and other road operators, which are seen as the most efficient and reliable source for update information, are at the beginning of the chain. Processes to define, install and change safety road attributes, like issuing of regulations and work orders, are not included in the scope of the project. ROSATTE looks only at the outcome of these processes, and especially at incremental changes in attributes. For this, it will, on the data provider side, study database storage of attribute information and update mechanisms, and methods for extracting the information, both complete data sets and incremental updates.

The extracted information is the input for the data exchange infrastructure, which is the topic of this document. On the user side, the project will study data integration, especially at providers of digital map databases for navigation systems and other driving assistance applications. The data chain from map providers to in-vehicle systems, which is depicted in the lower part of Figure 1, is not part of the project scope. Of this chain, especially the

part representing incremental updating of the in-vehicle map database at regular, short intervals, will benefit from data chain that ROSATTE intends to realize while at the same time providing a rationale for the project.

Setting up this chain has a clear benefit for public authorities and road operators through its potential contribution to improving road traffic safety, while giving the industry the opportunity to improve the quality of map databases used in in-vehicle systems, and enabling new safety applications that need map data with Europe-wide complete and up-to-date coverage of road safety attributes.

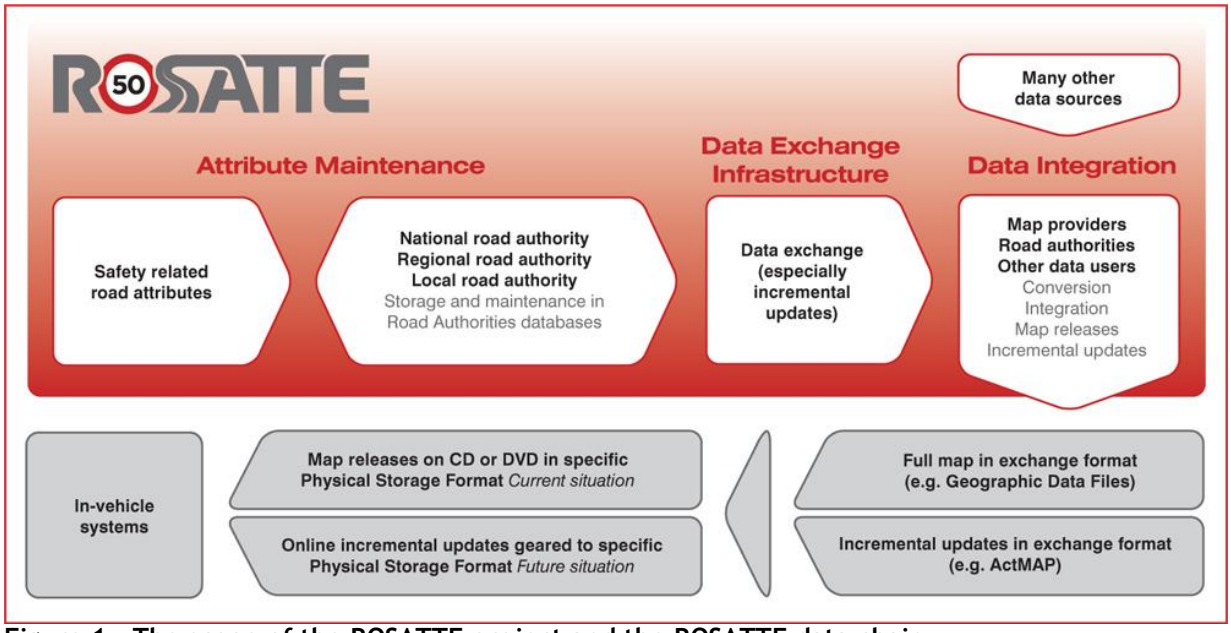



**Figure 1 - The scope of the ROSATTE project and the ROSATTE data chain**

# Excutive Summary

This document specifies the general outline of the data exchange methods. These specifications are necessary to enable automatic and timely exchange of safety attributes between road authorities and potential users of such data. Moreover, it will provide reference implementations for the exchange of data during test and validation period.

In a first step, the relevant requirements identified from the deliverable D1.2 are identified and summarized, in addition the service requirements are prioritized according to their significancy.
Secondly, a selection of viable technologies for the data exchange is defined based on relationship between specification elements and requirements.

The core part of the document provides the specification of a mechanism for data exchange of road safety information. It provides especially:

- A conceptual specification of the data content (information model). This is done using UML (packages, class diagrams, attributes, associations and OCL constraints). The data content specification is organized in a number of packages wehre each package correspons to a separate subset of the ROSATTE domain.
- A physical exchange format (structure and coding using GML schema) to specify a coding for the various types of data listed under the conceptual model.
- A service specification is implemented using UML (class diagrams), in order to facilitate the actual data flow between the various actors within ROSATTE. This service specification is inspired by INSPIRE network services architecture.

After full consolidation of these specifications, the data exchange specification will be given, which implements the conceptual specification as xml-schema definition (.xsd) and an implementation of the service specification as web-service will be provided. Currently an implementation specification according to REST [34] is defined, see table 10 in section 6.4.2. This does not exclude the option of future implementations based on other technologies such as SOAP [29].

In annex A, the concrete schemas (xsd) defining the ROSATTE physical exchange format is presented.

# Table of contents

## List of tables



## List of figures

# 1. Introduction

## *1.1. ROSATTE actors and high level use cases*

A first high level description of the ROSATTE system and data chain has been established in the ROSATTE Deliverable D1.2 'Requirements and Overall Architecture'. The data chain in ROSATTE includes three main use cases in which the ROSATTE actors are involved (see Figure 2 later in this document).

'**Maintain attributes**:' Enacting Authorities (or road operators, in ROSATTE called Road network managers), which are involved in deciding or implementing safety related aspects of roads (e.g. speed limits, overtaking bans or pedestrian crossings etc.) need to have road safety related data maintained to support their internal systems and processes. Technically these operations of a road safety data store may be done by a separated entity which we call 'Data store operator'. Such data maintenance is usually done without having the exchange of such data with external parties in mind. Such data maintenance can be done in the context of devising traffic regulations (traffic regulation data base) or in the context of road infrastructure maintenance (traffic sign databases). In addition, field surveys of road features and equipment may be a source for such data. If data shall be exchanged compliant to ROSATTE exchange specifications, the data store has to fulfill certain requirements. ROSATTE Deliverable D2.1 'Conceptual specification on how to establish a data store' describes the use cases and processes, which make road data maintenance operations compliant with a ROSATTE conform data exchange.

'**Exchange attributes**': This use case deals with the exchange of road safety related data between Enacting Authorities and their Data Store Operator (or Road Network Managers, depending who maintains the data) and 'information providers' (i.e. map makers), which are responsible for integrating safety related data with other data in order to provide it to service and application providers. ROSATTE creates a harmonized way for this data exchange. This document is about specifying this way to exchange road safety related data. By a widespread usage of this data exchange mechanisms specified here, a harmonization can be achieved, greatly facilitating the efficient usage and the quality of road safety data for the information providers.

'**Integrate attributes**': Here, the Information Providers integrate the data received according to the ROSATTE data exchange specification into their own database for further distribution to their customers (service and application providers).

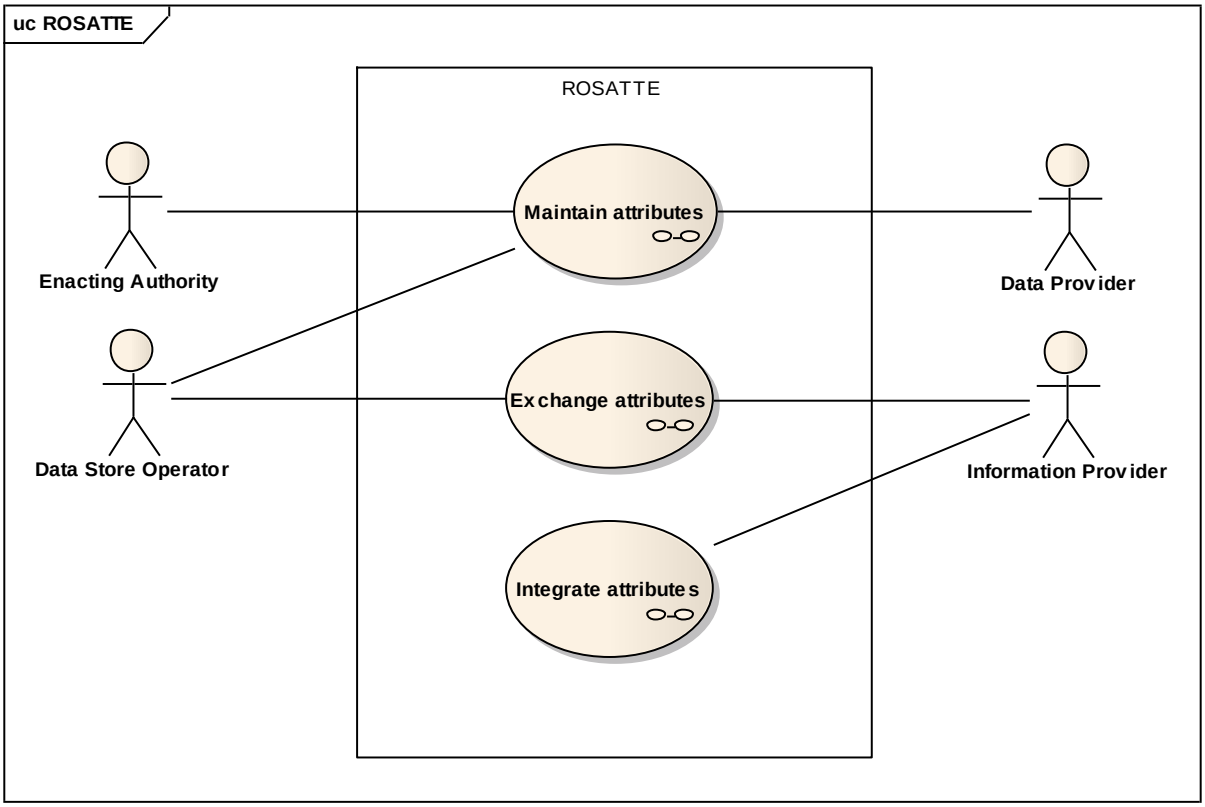



**Figure 2 - Main use actors and use cases in ROSATTE (from ROSATTE Deliverable D1.2 Requirements and Overall Architecture V1.1, 28.8.08)**

## 1.2. ROSATTE data exchange

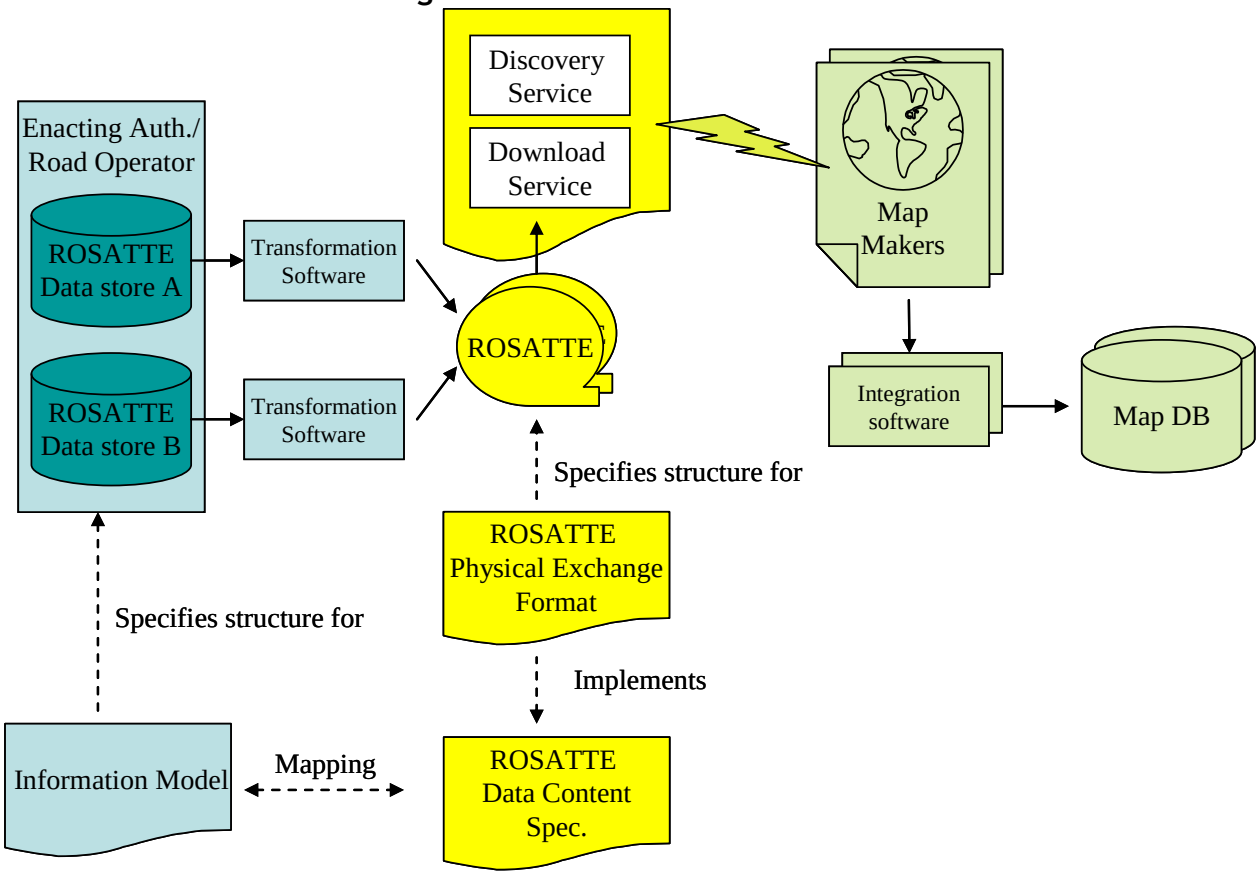



**Figure 3 - conceptual and concrete level of ROSATTE data exchange**

Underlying to the ROSATTE data exchange is an information model (or data content specification), which is the conceptual schema according to which data transferred according to ROSATTE can be correctly interpreted and understood. With regards to the road safety information, this information model defines amongst others:

- How ROSATTE exchange conceives a road safety feature;
- How ROSATTE exchange describes geographic location;
- How ROSATTE exchange describes full initial data supply or updates.

Figure 33 depicts the conceptual explanation of the data exchange process, if a data store on the enacting authority/road network operator's side shall supply data in a ROSATTE compliant way, its internal information needs to be mapped to the ROSATTE information model on a conceptual level.

The specification of the physical exchange format then represents a translation of the ROSATTE information model into an implementation specification suitable for data exchange.

Conceptual mapping and the physical exchange format specification then allow to conceptually and physically transforming data from the enacting authorities/road operators on the supply side (ROSATTE data stores) into a ROSATTE data set for exchange with the information providers.

The ROSATTE compliant data set can then be requested by a client/map provider on the receiver side and downloaded through a service over the web.
The data set is then available on the information provider's side for integration into its data base.

With regards of the scope of this data exchange, the following aspects are important:

- ROSATTE data exchange is not restricted to a particular type of safety feature.
    - The underlying conceptual data content model is designed in a generic way which is easily extensible by adding further safety attribute type values. Such an extension does not lead to an incompatibility with a previous version/implementation of the interface.
    - The conceptual data content model allows including point, line and area related features.
- This includes safety features such as speed limits, overtaking bans, access restrictions etc.
- ROSATTE data exchange specification focuses on the service level on the download and feedback services. Services on data discovery are explicitly excluded, since the discovery services as promoted by the INSPIRE Directive will include ROSATTE type service discovery as one theme. Secondly, in the context of the present research and demonstration project, service discovery (amongst cooperating partners) is not a relevant subject.

## *1.3. Location referencing*

The safety attribute information to be exchanged is related to real world locations on the road network, and needs to be referenced to this network. For this a robust location referencing method is needed, which permits encoding, transfer and decoding of the location information with a high degree of certainty. As any part of the network should be addressable, the location referencing method to be used should be flexible and dynamic. For this reason, map-based dynamic location referencing is the preferred approach. Currently the best available method for this is the AGORA-C location referencing method (AGORA-C 2006a), which is incorporated in the ISO International Standard "Location referencing for geographic databases", Part 3 "Dynamic Location References (Dynamic Profile)" (AGORA-C, 2007). In tests, this method has shown a hit rate (percentage correctly matched plus percentage correctly identified as not present in the map database of the receiving side) of > 98% (AGORA-C, 2006b). Parts of the method are protected by IPR, and licensing conditions for the method have not yet been published. Due to this and dependent of the actual future licensing conditions, wide-spread adoption of the method may be severely hampered. This may imply that a future operational implementation of the ROSATTE approach may need an alternative method for location referencing. Despite this, for the current ROSATTE research activities the AGORA-C method has been judged as the best available option.

## *1.4. Structure of the document*

The structure of the document is as follows:

- In chapter 1, the project description, providing essential background on the project
- In chapter 2, the objectives and scope of this document

- In chapter 3, definitions and acronyms
- In chapter 4, an introduction providing the basic concepts of ROSATTE
- In chapters 5 and 6, a summary of requirements and the selection of conformant technologies
- In chapters 7, 8 and 9, the exchange specification itself consisting of
    - a data content specification,
    - a physical exchange format specification, and
    - a service specification.
- In annex A, the concrete schemas (xsd) defining the ROSATTE physical exchange format

# 2. Definitions and acronyms

## *2.1. Definitions*

| Term | Definition |
|---|---|
| (Legal) Traffic regulation | Legal order established by an enacting authority, which regulates the use and equipment of roads e.g. with regards to speed, overtaking ban, traffic lights etc. at a specific location. It often leads to the installation of *traffic signs* at this location.<br>The traffic regulation may cover a point along the road (e.g. pedestrian crossing), a linear location (speed limit along the road from location A to B) or an area location (30km/hour zone, i.e. a set of streets in an area) |
| Traffic sign | Signs (e.g. speed limit signs) which are put up by road maintenance operators as a manifestation of a traffic regulation for driver information. Traffic signs, by their nature are point objects. To describe a line or an area regulation, several traffic signs may be put up for clarity of information to the drivers.<br><br>Traffic signs data are often maintained as a part of separate data bases by road maintenance authorities in order to more efficiently manage road (side) equipment. While for certain signs, the content is close in meaning to the corresponding 'safety attribute', transformation is needed to create a safety attribute together with a correct location description. E.g. several traffic signs (repeated speed limit signs) may need to be analysed to define the location/extent of the corresponding safety attribute (speed restriction for several kilometers along a road). |
| Field survey | Capturing of information by physical road inspection. Measurements/recording of road features including traffic signs etc. on the roads in a specific area or for a specific road level.<br>Field surveys can be done as (repeated) total surveys of areas (full supply) or more ad hoc surveys from small areas.<br>Field surveys may be useful for safety attribute data bases e.g. when creating an initial data set on safety attributes or to support quality mechanisms by providing reference data on safety attributes, which come from other sources. |
| Road network database | Digital description of road network including certain attributes. There is different ways to describe a road network by data objects. In this document we mean the representation of roads by their geometry (holding a direct location description) and topology using segments and nodes to describe road sections and junctions. In cases where linear referencing methods are used the road network database the data needed for the definition of linear referencing systems may, but is not required to, be part of the road network database.<br><br>In order to be useful as a basis for on-the fly referencing methods, a road network databases has to include certain structures and attributes (called 'location referencing attributes' in this document) required for all road network databases between which on-the-fly references shall be exchanged. |

| Term | Definition |
|---|---|
| Safety Feature | Feature/attribute in a digital road database which describes the content of a traffic regulation. To be useful, each safety attribute along a road must be paired by the description of its location. The location may be a point, a linear or an area location.<br><br>To describe the location of a safety features/attributes it can be 'attached' to the road network by (logical) reference to the road database objects in order to clarify their location. Alternatively, a direct location description by coordinates is often used (geo-reference).<br><br>Its details (as well as the location information) may be directly derived from a traffic regulation (or it could hold a reference to the regulation at its origin). Alternatively, its details (and location information) may be captured by field survey, or from databases including traffic signs.<br><br>Note:<br>In ROSATTE data stores at enacting authorities or in the digital maps of the information- or map providers, ROSATTE data may be represented either as separate features associated with locations at the road network or as attribution of the road network itself. The term "Safety attribute" in documents D1.2 [18] and D2.1 refer to both the stored data at either end of the exchange and the exchanged data itself. The term Safety feature in D3.1 refers specifically to the representation of the data which is being exchanged in ROSATTE using the exchange specification. Therefore the terms may therefore be viewed as synonyms since it is the same real world entities that are being represented. |

| Term | Definition |
|---|---|
| Location referencing | Location referencing describes a method to describe a location of an object in a digital database.<br><br>Direct Location referencing uses a description of a location by geo-references (i.e. a description in reference to a geodetic reference system, e.g. latitude and longitude coordinates in WGS84). In the document, geo-references are often used synonymous to direct location references.<br><br>Indirect location referencing describes a location by its logical reference to other objects (e.g. road segments, or nodes) in a digital database, which themselves hold direct geo-references for describing their location. In our case these other objects are those of the road network database (segments, nodes). Indirect references to the road network will also be called 'network reference' in the document.<br><br>Indirect location references can usually only be interpreted in connection to the specific database objects, which they point to (logical reference).<br><br>If location references shall be independent of one database, there exist two main ways:<br>❖ "[It] can be implemented by pre-coding often used locations, as is done in RDS-TMC. The location codes and related additional information are stored in so-called location tables. Advantage is the conciseness of the code; disadvantage the limited amount of addressable locations." (From the AGORA Specification [1], [2] and [3]). These predefined and coded locations need to be integrated in each network database, between such location references shall be transferred.<br>❖ On-the-fly location referencing, which is another method to identify a location independently from a specific instance of a geographic information database. A location known in ones own geographic information set (as indirect location reference) is transformed by an encoding mechanism (using certain rules) into a location code, which can be decoded in reference to another geographic information set (as indirect location reference). The location code contains direct location referencing information (usually geo-references to point locations) as well as typical features/classifications of the (indirectly referenced) data base objects (e.g. classification of a road segment, street names or the like), which are common to all databases.<br><br>The location referencing method used creates requirements for the source and the target geographic information set with regards to structure and content. |
| Linear Referencing | The Linear Reference System (LRS, also called Linear Referencing System) is a reference system in which features are localized by a measure along a linear element. Each feature is localized by either a point known as a "milepoint" or a linear event ("segment"). The system is designed so that if a segment of a route is changed only those milepoints on the changed segment need to be updated. (From Wikipedia).<br><br>In the above classification, LRS is an indirect location referencing technique, which –in the case of a road network – uses 'routes' (a directed chain of segments) as 'aggregated' linear objects, in reference to which a linear location is described either by distance measures (from km 115 to km 120 in reference to the start point at km 0) or percentages (from 55% to 58% of the route length). |

| Term | Definition |
|---|---|
| AGORA | AGORA [1], [2] and [3] is one on-the-fly location referencing method, which is made reference to throughout this document.<br>AGORA requires a digital network description that includes (1) geometric road information, (2) which has a topology to allow routing functions and (3) which contains certain road attributes, such as 'form of way' and 'functional road class'.<br><br>AGORA was initially developed in an EU funded research project and has evolved into an ISO standard. |
| Full supply of data | Supply of data with exhaustive comprehensive coverage. A full supply of data is needed when 'initialising' a database, i.e. filling it comprehensively with the data in question. A full supply of data may be established in regular intervals, i.e. asynchronously of data changes. It then replaces the previous data set. The up-to-dateness of the database is only given at moments, where a (new) full supply is available.<br>Road network data are typically updated by full supplies in regular intervals. |
| Incremental supply of data | Method for regularly updating a database as soon as changes occur in data. The up-to-dateness of the database is continuously maintained.<br><br>Incremental updates can be provided either through record update events or by comparing different database states. In the case of record update events it is necessary to store information that describes when a safety object is added, modified or deleted. The other option is to send updates as batches of increments created from a process comparing the latest version of a dataset to a previous one. |

## *2.2. Acronyms*

| Acronym | Definition |
|---|---|
| UML | Unified Modelling Language (http://www.omg.org/uml) |
| e-commerce | **Electronic commerce**, commonly known as (electronic marketing) **e-commerce** or **eCommerce**, consists of the buying and selling of products or services over electronic systems such as the Internet and other computer networks [http://www.wikipedia.org] |
| GeoRM | **(Geo Rights Management)** technology to allow public authorities to electronically specify licence terms and conditions in such a way which supports the automated transfer of legal rights to use the spatial data or service; Services to enable e-government integration of network services, to manage authentication, authorisation, pricing, billing, logging and so on. [INSPIRE Network Services Architecture] |
| DOM | Document object model. A method for access of HTML and XML documents that typically relies on loading the entire document into memory where the client accesses the data using an API. This method offers simplicity when accessing data but does not scale well to large documents since memory still is a rather limited resource. |
| SAX | Simple API for XML. An event-driven method for sequential access of XML documents. This is an alternative method to DOM with the advantage that it typically scales better to large xml documents since it allows for a "process and forget" approach that processes the document piece by piece instead of all at once. |

| Acronym | Definition |
|---|---|
| ADAS | Advanced driver assistance systems |
| Base64 | The term Base64 refers to a specific MIME content transfer encoding. It is also used as a generic term for any similar encoding scheme that encodes binary data by treating it numerically and translating it into a base 64 representation. [http://en.wikipedia.org] |
| OSF/DCE | Open Software Foundation/Distributed Computing Environment |

# 3. Background

## *3.1. Safety features - scope and definition*

### *3.1.1. Scope*

The MAPS&ADAS sub-project of the Integrated Project PReVENT focused on two main topics: (1) the ADAS (Advanced Driver Assistance Systems) Interface Specification, defining a protocol for the exchange of ADAS relevant map information in the form of an ADAS horizon, from an ADAS horizon provider (AHP) having direct access to the map database, over a vehicle bus to distributed ADAS applications; (2) data sourcing of map attributes for safety related ADAS applications.

In the framework of the data sourcing activity, a list of eight priority attributes was defined. This list is summarized in Table 1 ordered by priority, from highest to lowest.

**Table 1 - Overview and description of priority safety attributes (MAPS&ADAS, 2004) [5]**

| safety attribute | Description |
|---|---|
| legal speed limit | legal maximum speed limit, especially static or permanent, and either general (implicit, not necessarily signposted) or specific (explicit, signposted), and either fixed (always one and the same value) or variable (time or weather dependent) |
| traffic sign | especially hazard warning signs and priority signs (and excluding specific speed limits as these are treated separately) |
| lane information | number of lanes, lane width, divider characteristics between lanes, longitudinal connectivity |
| traffic lights | presence of traffic lights at intersections |
| Crossing | crossings with other traffic modes, especially pedestrian, bicycle, tram and railway crossings |
| accident hot spot | locations that are identified as potentially dangerous based on accident statistics plus a specification of these circumstances (speed, position, probable route of the vehicle, date, time, temperature, light condition, weather, etc.) |
| Slope | longitudinal road gradient |
| Banking | transverse road gradient or superelevation |

This list was extensively discussed within the framework of MAPS&ADAS, and in relation to this with both industry partners and public authorities. From these discussions a priority ranking has resulted which coincides with the ranking order as indicated in Table 1.

Three of these attribute types, **speed limits**, **traffic signs** (other than speed limit signs) and **traffic lights**, may be roughly categorized as signpost related attributes, as they relate to a signpost positioned along the road. This is especially true for specific speed limits and other traffic signs, but in a way also for traffic lights. Each of these are rather easily introduced, changed or removed, although introduction or removal of a traffic light takes more effort than placing a speed limit or other traffic sign. Changes may occur frequently, and only require a low preparation, planning and implementation effort on the side of authorities.

**Lane information** and **crossings** with other modes are related to the road infrastructure, and are less easily changed than signpost related attributes, i.e. changes will not occur

very frequently, and a change needs a medium preparation, planning and implementation effort, except for tram and railway crossings, for which the effort may be considered high.

**Slope** and **banking**, being characteristics of the road itself, are more intimately related to the road infrastructure and layout. Changes in these attributes as well as in tram and railway crossings will occur rather infrequently, and such changes need a high preparation, planning and implementation effort.

**Accident hot spots** or accident risk locations constitute a separate attribute category, as they are not instantaneously measurable on the road. The attribute is either derived from accident statistics, which are kept by road authorities, or based on identified relationships between road characteristics and accident risk. Both sources may be used in combination. (MAPS&ADAS, 2007) [6]

An extended and slightly modified version of Table 1 is included in the ROSATTE Technical Annex (ROSATTE, 2007) [9], and reproduced in Table 2 below.

**Table 2- Safety attributes for in-vehicle ADAS applications (ROSATTE, 2007)**

| Safety Attributes | Example of ADAS application | Change frequency |
|---|---|---|
| Speed limit | Speed alert | Very high (7-9% / year) |
| Traffic signs | Enhanced navigation (e.g. truck) | High |
| Lane information (number, width, divider, connectivity) | Lane keeping assistance, Lane departure warning, Curve warning | Medium |
| Traffic lights | Intersection assistance | Medium |
| Crossings (pedestrian, tram) | Enhanced navigation, Vulnerable road-users protection | Medium |
| Toll barriers, motorway junctions, tunnel access | Obstacles / change of lighting / speed limit / inter-vehicle distance management | Very low (new road or reshaping) |
| Gradient (slope) | Curve warning, Fuel consumption assistance (car and truck) | Very low (new road or reshaping) |
| Transverse gradient (banking) | Roll-over warning system (truck), Curve warning | Low (new road or reshaping) |

Based on the list of safety attributes described in Table 1, it was agreed by data providers (mainly road authorities as data owners) and data users (map providers, vehicle industry and road authorities) that the ROSATTE project needs to address the attributes with the highest priority market demand and focus on a limited number of safety attributes characterized by a high change frequency (ROSATTE, 2007) [9]:

- speed limits;
- other traffic regulations and signs relevant to road safety.

This implies that the focus in the ROSATTE project is on safety road attributes that may be expressed by traffic signs or road markings. However, explicit signage will not always occur, e.g. for implicit or general speed limits, like the general speed limit that applies for all motorways in a country unless replaced by an explicitly signposted speed limit, and the general speed limit for built-up areas, which is only signposted (but not always with a value) at the entrance of the area, and nullified when leaving the area. Turn restrictions may be indicated by signs, road markings or both.

### 3.1.2. Traffic regulations and signage

Traffic signs and road markings are expressions of traffic regulations. A specific traffic regulation may be defined as a directive involving an instruction for road traffic related behavior for a specific location on the road network, made and maintained by the public authority that is responsible for the road network to which the location belongs. A specific traffic regulation in this sense is based on an official document, and may be expressed on the road by traffic signs and/or road markings. General traffic regulations may be defined as general instructions for road traffic related behavior that apply to the network as a whole, and are based on legislation and legislation based orders.

Specific regulations are the basis for the signage on the road (traffic signs and road markings). The result is a certain feature at a certain location. It is especially this feature that is relevant for the ROSATTE data exchange, not so much the traffic regulation, or the traffic sign and its precise location or the road marking. The feature has a type and, for certain types, additional values, and may have for most types additional descriptive information as expressed on sub-plates of traffic signs.

### 3.1.3. Coding of road safety features

The road safety features that are relevant in the context of ROSATTE can in principle all be expressed by traffic signs and road markings. Each of these features needs to be coded. It seems feasible to revert to the Vienna Convention for a coding scheme. The following three paragraphs were taken, mutatis mutandis, from (MAPAS&ADAS, 2007) [6].

In Europe traffic signs or road signs are internationally standardized by the "Convention on road signs and signals" (UN, 1995) [13], which is often referred to as the "Vienna convention on road signs and signals" or shortly just the "Vienna Convention". This international treaty was established as the result of an international conference held in Vienna in 1968. The current version is the amended version that entered into force in March 2006. A related treaty that was a result of the same conference in 1968 is the "Convention on road traffic" (UN, 1993) [12], which is often referred to as the "Vienna convention on road traffic" or also shortly as the "Vienna Convention". The latter term therefore may refer to either of the two treaties.

The "Convention on road signs and signals" (UN, 1995) [13] (here further called the "Vienna Convention") defines Road signs (Chapter II), Traffic light signals (Chapter III), and Road markings (Chapter IV). Road signs are divided in eight categories: Danger warning signs (A), Priority signs (B), Prohibitory or restrictive signs (C), Mandatory signs (D), Special regulation signs (E), Information, facilities or service signs (F), Direction, position or indication signs (G), and Additional panels (H). In the MAPS&ADAS list of recommended priority attributes the term "traffic signs" relates especially to regulatory signs which are covered by categories A - E.

Note that the Vienna Convention uses the term road sign, while in the context of MAPS&ADAS and in the GDF document (GDF 4.0) [4] the term "traffic sign" is used. The two terms have identical meaning. In addition, also the term "road traffic sign" is found. The Vienna Convention does not explicitly define a road sign, but implicitly it is defined as: "A sign for signifying a certain rule or conveying certain information to road-users." Another proper (and similar) definition is: "A sign near a road is giving information or instructions to drivers." Oxford, [8].

## 3.2. Location referencing

### 3.2.1. Introduction

The information to be exchanged in the context of ROSATTE has two core elements: attribute feature and location, and in this respect much resembles traffic information messages, in which traffic information (event or status) is exchanged for well-defined locations on the road network. Location referencing is the term used for technology to describe a location on the road network in a standardized and coded way.

### 3.2.2. Milestone systems

Milestone (or milepost) systems are in fact location referencing systems, and were first used by the Romans (milliaria). More refined versions of such systems are still in use today. In this approach, roads are considered as long linear, more or less independent, entities without explicit mutual topological relationships. Locations along the entity are expressed in a one-dimensional location referencing system, using positions along the entity. In GDF this approach is named chainage referencing. (GDF 4.0).

### 3.2.3. Pre-coding of locations

RDS-TMC is a system for providing traffic information messages via the Traffic Message Channel (TMC) on the Radio Data System (RDS), a side band of the Frequency Modulation (FM) radio system used for data broadcast. For this system a specific location referencing method was developed (since the late 1980's), that uses pre-coded locations, and which is described in Part 3 of the TMC Standard (EN ISO 14819-3) [11]. Location codes having a size of 16 bits are stored in a location table together with additional information about the location. All the three location types (point, linear and area) are used. The location table has a hierarchical structure, with point locations referring to linear locations, and linear locations to other linear locations and area locations. The area of type country or continent is at the top of the hierarchy. In practice only point locations are used for referencing traffic information. Due to the limited number of available location codes per table, and due to costs for creation and maintenance of the tables, only a limited part of the network is generally addressed. As point locations are generally chosen at intersections or connections with other roads, different solutions have been developed to refine the original location referencing method. Among them one can mention the use of relative distances to the considered points (as done in DATEX and DATEX II) or a proposed method for precise TMC location referencing which has been developed in TMC Forum as an extension to TMC [10]. TMC location coding was originally developed for textual messages, but has since its introduction in 1998 mainly been used with navigation systems.

### 3.2.4. Map-based location referencing

Another possibility is to use information from the map database to construct a location code. Such map-based method could work ad-hoc or dynamically. A code is constructed when needed, used in a message, decoded at the receiver side, and then discarded. Main advantages are that no location tables are needed, that the whole network is addressable, and that any arbitrary location in the network can be coded. Ad-hoc or dynamic location referencing uses geometry and attribute information. An adequate method needs to be able to overcome differences between map databases. This means that generally some redundancy in attribute content is needed to resolve difficult cases. The AGORA-C method was developed following work in the AGORA project, which followed earlier work in the EVIDENCE project. It was rather extensively tested, showing good results [AGORA, 2006b],

and was published as a specification [AGORA-C, 2006a], which was a major input for ISO standardization. In November 2008 the related standard was adopted [AGORA-C, 2007]. AGORA-C was developed for traffic information and other telematics messages. AGORA-C was specifically developed for exchange of location codes for GDF type map databases, and therefore makes especially use of GDF specific map attributes. This may be a drawback if the public authorities use non-GDF map databases or no map at all.

## *3.3. Update types*

Besides exchanging data about safety features as such, ROSATTE needs to define a clear way in which only the incremental updates for safety feature data can be exchanged. The reason for this is that it will make data exchange as efficient as possible.

Different update types can be distinguished. A classification and descriptions are provided in Table 3.

**Table 3 - Update types and update type descriptions**

| update type | Description |
|---|---|
| insert | a new safety feature is inserted at the referenced location, without affecting any existing safety feature |
| modify | modifies the attributes of an existing safety feature referenced by an object identifier |
| remove | removes the existing safety feature referenced by an object identifier |

The above listed update types all require the use of permanent object identifiers by which safety features can be unambiguously identified. A permanent object identifier is an identifier which is unique within the scope of and follows a safety feature during its entire lifetime. Some public authorities may use such identifiers already, which in such cases could be used straight away.

In this specification and in the ROSATTE testing, it is proposed that the exchange of incremental updates is based on the use of permanent object identifiers.

## *3.4. Metadata*

Metadata for ROSATTE has the same role as for INSPIRE:

- basis for data discovery
- basis for data evaluation
- description of exchanged datasets

This document specifies metadata for these purposes based on the INSPIRE implementing rule for metadata.

# 4. Requirements

## *4.1. Requirements from D1.2*

The table below lists the requirements on ROSATTE that were identified in the deliverable D1.2 [18]. The items that are grayed in the table below are considered to concern other work packages of ROSATTE and are not considered for this specification.

Explanation to the ID column:
- FR - functional requirement
- NFR – non-functional requirement
- CR – conformance requirement

**Table 4 - Requirements from D1.2**

| ID | Requirement name | Short definition | Priority:<br>C: Critical<br>S: Significant<br>I: Of interest | Comments, links to other requirements, open issues |
|---|---|---|---|---|
| FR-1 | Data discovery | A specification of a Discovery service with metadata shall be available. | C | The ROSATTE infrastructure shall provide discovery service with suitable metadata, that enables the Information Provider to easily find services providing road safety attributes.<br><br>Note: Within ROSATTE WP3, due to time constraints, the discovery service has not been prioritized. The INSPIRE implementing rules for discovery services are believed to be valid also for ROSATTE. |
| FR-2 | Standardized access | Data Services and their use shall be specified. | S | No matter what the content is, accessing and using Data Service's is done the same way across the Europe. Guidelines stating how to access Data Services in a standardized way shall be defined. |
| FR-3 | Data subscription | Guidelines specifying how to subscribe to road safety attributes in the ROSATTE exchange infrastructure shall be provided. | S | Information Providers can subscribe for change notifications for their individual needs. Data subscription functionality is created by combining with FR-5. |
| FR-4 | Specification of Quality management procedures | Guidelines specifying how to quality assure received road safety attributes shall be specified. | C | Guidelines for automatic and semi-automatic quality check routines must be specified. These guidelines should be incorporated into existing procedures. |
| FR-5 | Incremental updates | The ROSATTE infrastructure shall provide both incremental updates and full updates of road safety attributes. | C | Incremental update datasets can be defined using received change notifications. |

| ID | Requirement name | Short definition | Priority:<br>C: Critical<br>S: Significant<br>I: Of interest | Comments, links to other requirements, open issues |
|---|---|---|---|---|
| FR-6 | Unambiguous location referencing | The road safety attributes provided through the ROSATTE infrastructure shall be structured to enable unambiguous decoding and interpretation of the referenced locations. Different locating methods allowed. | C | |
| FR-7 | Data Store initiation | The project shall provide guidelines for Data Store design and initiation. | I | |
| FR-8 | Data import | The project shall define guidelines for import of road safety attributes and road network data. If suitable import tools are non-existent, new tools shall be developed. | S | |
| FR-9 | Workflow support | The project shall produce a specification of tools and guidelines for integrating data maintenance with legal workflow. | I | |
| FR-10 | Presentation and maintenance tools | The project shall develop specifications of how to present and maintain the road safety attributes. If existing tools are not suitable, new tools shall be developed. | C | Where suitable existing tools do not exist, tools that enable presentation, maintenance and publishing of the road safety attributes must be developed and implemented. The tools shall be built on existing work and standards where suitable. |
| FR-11 | Feedback loop | A feedback channel from information providers back to enacting authorities shall be provided. | C | |
| FR-12 | Integration tools | Tools to integrate road safety attributes into existing information providers systems shall be developed if existing tools does not provide the satisfactory functionality. | S | Generic software components must be developed to integrate the road safety attributes in a quality assured (ref FR-4) and automated manner. |
| FR-13 | Flexible type definitions | The meaning of “Road safety attributes” is not finally decided. Changes will occur in the future. | C | It shall be possible to add and change (to some degree) the available type definitions describing road safety attributes. |

| ID | Requirement name | Short definition | Priority:<br>C: Critical<br>S: Significant<br>I: Of interest | Comments, links to other requirements, open issues |
|---|---|---|---|---|
| NFR-1 | Availability | Valid quality parameters related to availability shall be declared in the metadata associated with the road safety attributes. | C | Degree to which geographic data is available at a certain place and at a defined time. Possible quality parameters:<br>- Communication failure rate<br>These quality parameters should be specified in the metadata attached to delivering system. |
| NFR-2 | Up-to-dateness | Valid quality parameters related to up-to-dateness shall be declared in the metadata associated with the road safety attributes. | C | Degree of adherence of geographic data to the reality changing with time. Possible quality parameters:<br>- Date of last update<br>- Date of origin<br>- Rate of change |
| NFR-3 | Completeness | Valid quality parameters related to completeness shall be declared in the metadata associated with the road safety attributes. | C | Degree of availability of all information needed to describe the reality. Possible quality parameters:<br>- Missing data<br>- Surplus data |
| NFR-4 | Correctness | Valid quality parameters related to correctness shall be declared in the metadata associated with the road safety attributes. | C | Degree of accordance of geographic data (feature(s), attributes, functions, relationships) to corresponding elements in reality, up-to-dateness being presumed. Possible quality parameters:<br>- Geometric correctness<br>- Topological correctness<br>- Thematic correctness |
| NFR-5 | Consistency | Valid quality parameters related to consistency shall be declared in the metadata associated with the road safety attributes. | C | Degree of accordance of geographic data (data structure, their features, attributes and relationships) to the models and schemas (conceptual model, conceptual schema, application schema and data model).<br>- Geometric consistency<br>- Topological consistency<br>- Thematic consistency |
| NFR-6 | Accuracy | Valid quality parameters related to accuracy shall be declared in the metadata associated with the road safety attributes. | C | Degree of adherence of geographic data to the most plausible or respectively the true value.<br>- Absolute position accuracy<br>- Relative position accuracy<br>- Quantitative attribute accuracy |

| ID | Requirement name | Short definition | Priority:<br>C: Critical<br>S: Significant<br>I: Of interest | Comments, links to other requirements, open issues |
|---|---|---|---|---|
| NFR-7 | Reduced data update delay | The time delay from the moment a Public Authority regulation is effective, until the end user data have been updated, shall be reduced. | S | Related to NFR-2. The infrastructure it-self may have minor delays, but the ad-ministrative routines on public authority side must be adapted to the lifetime of the data handled. Update frequencies of 24 hours or less is a reasonable require-ment. |
| CR-1 | Conformance with European law. | The ROSATTE infrastructure shall offer its services in a way that conforms with the INSPIRE directive. This in-cludes creation and mainten-ance of metadata, a discov-ery service using it with a minimum set of search crite-rions, view services, down-load services and supporting services. | C | SOA Web Services<br>Minimum metadata elements required:<br>Identification (Name,type,URL)<br>Classification<br>Keyword<br>Geographic location<br>Temporal reference<br>Quality and validity<br>Conformity.<br>Access conditions<br>Access limitations<br>Responsible organization<br>Minimum search criteria:<br>Classification<br>Keywords<br>Geographical location<br>Quality and validity<br>Access conditions<br>Responsible organization<br>**Links to functional requirement FR-1** |

## *4.2. Service requirements*

In the following requirements are set up with regards to the service for downloading safety features from ROSATTE data stores on the public authority/road operator's side.
Requirements qualified as 'critical' or 'significant' appear essential for prototype implementation within the projects, those qualified as 'of interest' may be ignored at this stage of implementation.

**Table 5 - Service requirements**

| ID | Requirement name | Short definition | Type<br>F: Functional<br>N: Non-functional<br>C: Context | Priority:<br>C: Critical<br>S: Significant<br>I: Of interest | Link to user req. | Comments, links to other requirements, open issues |
|---|---|---|---|---|---|---|
| R-1 | Download | A service which allows the download of update information about safety attributes has to be available. | F | C | | |
| R-2 | Download format | The format of the update information of all enacting authorities has to follow a common, well defined format. | F | C | | |
| R-3 | Update notification | It has to be possible to get notified about the availability of updates (push scenario) | F | I | | |
| R-4 | Feedback | It has to be possible to provide feedback information to the enacting authorities about the integration process of the provided update information | F | C | | |
| R-5 | Server side authentication | The origin of update information has to be secured | F | I | | This can be achieved using digital signatures. Certificates are needed for this. |

| ID | Requirement name | Short definition | Type<br>F: Functional<br>N: Non-functional<br>C: Context | Priority:<br>C: Critical<br>S: Significant<br>I: Of interest | Link to user req. | Comments, links to other requirements, open issues |
|---|---|---|---|---|---|---|
| R-6 | Download information integrity | The receiver has to be able to verify that the received update information has not been altered during transmission. | F | I | | This can also be achieved using digital signatures. Certificates are needed for this. |
| R-7 | Processing of large update data sets | The sender and receiver side has to be able to deal with large update data sets. | N | S | | This can be achieved using the SAX approach when dealing with XML data. It is discouraged to use the DOM approach as it will not scale well with the expected size of update data sets. |
| R-8 | Central instance | There has to be some sort of central instance which publicizes the network location of safety attribute providers. | F | I | | |

# 5. Selection of technologies

## *5.1. Introduction*

The state of the art document D1.1 [16] describes the current situation in 21 European countries and also Japan.

There are a vast amount of technologies that are referenced by D1.1. Some technologies go to a great depth and complexity. It is not possible for us in this document to describe all the detail of the referenced technologies and at the same time provide an overview. Therefore, a summary of the selection and the rationale is described here.

## *5.2. Relationship between specification elements and requirements*

The requirements have been listed in chapter 4. Roughly, the output from ROSATTE WP3 consists of three elements. These elements all concern the interface for data exchange between enacting authorities and information- (map-) providers. The elements are specified in the table below. The evaluated technologies contribute to one or more parts of the specification elements.

**Table 6 - WP3 specification elements**

| Name | Description | Applies to requirement |
|---|---|---|
| Data content specification **(DC)** | A conceptual specification of the content and structure of a ROSATTE dataset. This specification shall preferably use existing standards to the largest extent, both regarding the conceptual schema language and the data types used. The high level structure for the data specification is described in D1.2 [17] and contains the following parts:<br>- **Safety attributes** such as speed limit. Safety attributes are in this document called safety features.<br>- **Location references** and geometry. Ways of specifying the locations for the safety attributes<br>- **Update information**. Data (or actually a special kind of metadata) that in the case of incremental updates signal what actually occurred with the data described by the update information. May also contain transaction information (grouping of updates)<br>- **Metadata**. Describes the data. Includes data quality declarations.<br>The data specification may be merged with the data exchange specification below. There is however still a good reason to handle them separately to ensure that the conceptual data definition remains valid and reusable regardless of various physical formats. | **FR-5**, Incremental updates information can be a part of the data content depending of the chosen solution<br>**FR-6**, Unambiguous location referencing. The data elements needed must be specified.<br>**FR-11**, Feedback loop. Data that flows in the feedback loop must be specified<br>**FR-13**, Flexible type definitions<br>**NFR-1**, Availability<br>**NFR-2**, Up-to-dateness<br>**NFR-3**, Completeness<br>**NFR-4**, Correctness<br>**NFR-5**, Consistency<br>**NFR-6**, Accuracy<br>**CR-1**, Conformance with European law. The principles for a data content specification should not deviate from the preferred principles within INSPIRE. |

| | | |
|---|---|---|
| Data exchange specification **(DE)** | Defines the physical exchange structure for ROSATTE data. This includes the physical format (physical encoding of the concepts defined in DC) | **FR-5**, Incremental updates information can be a part of the data exchange depending of the chosen solution<br>**CR-1**, Conformance with European law. The physical format of ROSATTE should not deviate from the preferred within INSPIRE. |
| Service specification **(SS)** | A specification of the services needed at either side (enacting authority, information provider).<br>The service specification may define both the abstract interfaces and the concrete implementation technology. | **FR-1**, Data discovery. Interface needs specifying.<br>**FR-2**, Standardized access<br>**FR-3**, Data subscription. Interface needs specifying.<br>**FR-5**, Incremental updates. Depending on solution incremental updates may be handled in the service interfaces.<br>**FR-11**, Feedback loop. Interface needs specifying<br>**NFR-7**, reduced data update delay<br>**CR-1**, Conformance with European law. The services of ROSATTE should not deviate from the preferred principles within INSPIRE.<br>**R-1 – R-8** |

## *5.3. General assumptions regarding the review and selection of technologies*

As earlier stated, the amount of technologies referenced by D1.1 [16] is large and some of the technologies are quite extensive. To do this review from scratch would include reviewing and understanding tens of thousands of pages of specifications which in some cases are quite complex. To avoid this, we make the assumption that the implementing rules of INSPIRE serve as a technology baseline also for ROSATTE (ROSATTE requirement CR-1).

Below, a selected set of valid requirements and recommendations from referenced INSPIRE documents are listed.

### *5.3.1. Data content*

It is assumed that the requirements and recommendations from the INSPIRE document D2.5 [18] can be applied also to the ROSATTE data content specification. A selection of requirements and recommendations are listed below.

**Table 7 - INSPIRE requirements and recommendations valid for the ROSATTE data content specification**

| INSPIRE Requirement/Recommendation | Description |
|---|---|
| Requirement 4 | The reference model specified in ISO 19101 shall be used as the reference model of the INSPIRE data specifications |
| Requirement 5 | Every INSPIRE data specification shall conform to ISO 19131 |
| Requirement 8 | Every INSPIRE application schema shall contain a comprehen- |

| | |
|---|---|
| | sive and precise description of its spatial object types. |
| Requirement 9 | Every INSPIRE application schema shall conform to the General Feature Model as specified in ISO 19109 7.3-7.7. |
| Requirement 10 | Every INSPIRE data specification shall include one or more INSPIRE application schemas modeled according to ISO 19109 Clause 8, with particular attention to 8.2. |
| Requirement 13 | Spatial object types shall be modeled according to ISO 19109 7.1-7.2, 8.1, 8.5-8.9 and according to the additional rules in Clauses 9-12, 18, and 22 of this document. |
| Requirement 16 | Basic types as specified in ISO/TS 19103 6.5 shall be used in an INSPIRE application schema, whenever applicable. |
| Requirement 19 | Every INSPIRE application schema shall be specified in UML, version 2.1. |

### *5.3.2. Data exchange*

It is assumed that the requirements and recommendations from the INSPIRE document D2.7 [20] can be applied also to the ROSATTE data exchange specification. A selection of requirements and recommendations are listed below.

**Table 8 - INSPIRE requirements and recommendations valid for the ROSATTE physical format specification**

| **INSPIRE Requirement/Recommendation** | **Description** |
|---|---|
| Requirement 1 | Every encoding rule in INSPIRE shall conform to ISO 19118. In particular, it shall specify schema conversion rules for all elements of the application schemas to which the rule is applied. |
| Recommendation 2 | Encoding rules should be based on open standards.<br><br>GML (ISO 19136) and ISO/TS 19139 are promoted as the default encoding in INSPIRE. The main reasons for this are:<br>- GML and ISO/TS 19139 cover encoding rules for large parts of the INSPIRE application schemas. This is not the case for any other commonly used encoding.<br>- GML specifies a XML based encoding rule for ISO 19109 conformant application schemas specifying spatial object types that can be represented using a restricted profile of UML that allows for a conversion to XML Schema. In addition, GML provides a standardized encoding for many commonly used types from core standards of the ISO 19100 series (in particular ISO 19107, ISO 19108, ISO 19111, and ISO 19123) that form the foundation of the Generic Conceptual Model (D2.5).<br>- ISO/TS 19139 specifies a XML based encoding rule for conceptual schemas specifying types that describe geographic resources, e.g. metadata according to ISO 19115 and feature catalogues according to ISO 19110.<br>- The reference material provided by SDICs and LMOs shows that GML is increasingly used in Member States and international communities to represent and transfer geographic information.<br>- GML and ISO/TS 19139 are well integrated with the current candidate standards of the network services.<br>- The use of these standards is inline with the recom- |

| | |
|---|---|
| | mendations of CEN TR 15449 on encoding which promotes GML as the encoding method when transferring spatial objects and ISO/TS 19139 as the encoding method when transferring information related to spatial data such as metadata, feature catalogues and data dictionaries.<br>- A default encoding rule allows for a coherent encoding approach inline with the overall interoperability requirements of the Directive. |
| Recommendation 4 | For every INSPIRE application schema, a GML application schema should be specified. |
| Recommendation 5 | The encoding rule specified in ISO 19136 Annex E should be applied. For types within the scope of the ISO/TS 19139 encoding rule, the encoding rule of ISO/TS 19139 should be applied. |

#### *5.3.3. Service specification*

It is assumed that the requirements and recommendations from the INSPIRE document D3.7 [21] and D3.5 [22] can be applied also to the ROSATTE service specification. A selection of requirements and recommendations are listed below.

- In order to provide search support for all search metadata elements as defined by the INSPIRE Metadata IRs, the OGC CSW ISO 19115/19119 Application Profile (CSW ISO AP) shall be used as the reference specification for the INSPIRE Discovery Service [21].
- As the INSPIRE directive advises to utilize existing standards, OGC service bindings are taken as a guidance [22].
- In addition the World Wide Web Consortium (W3C) suggests the usage of SOAP as a messaging protocol for web services. INSPIRE services should utilize one standard technology binding for all service types. In order to streamline integration and implementation as well as getting a maximum benefit from the offered services, a mix of technologies is to be avoided. Taking all requirements, opportunities and risks into account, the default communication-protocol and binding technology for INSPIRE services should be SOAP [22].

### *5.4. Selection of technologies and rationale for selection*

#### *5.4.1. Overall criterions for selection*

The following table shows the criterions for selection of technologies in priority order:

**Table 9 - Criterions for selection of technologies**

| Nr | Criterion | Remark |
|---|---|---|
| 1 | Fulfill ROSATTE requirement | |
| 2 | Comply to European/International standard | |
| 3 | Comply with de facto standard/have a widespread acceptance such as the recommendations from the world wide web consortium. | |
| 4 | Positive contribution to ROSATTE specifications | If the top three criterions are not fulfilled, the technology may still make a contribution |

| | | |
|---|---|---|
| | | which is equal or better than something that is invented within the project |

If two or more technologies fulfill the same criterion the selection is based on a judgment on what technology fulfills the criterion in the best way.

### 5.4.2. *Proposed selection of technologies*

**Table 10 - Selection of technologies**

| Specification element | Selection | Rationale |
|---|---|---|
| Data content specification – conceptual schema language | UML [33] | Fulfills all criterions and is the conceptual schema language which is most widely accepted in the world today.<br>INSPIRE requirement |
| | ISO 19103 [24] | Fulfills all criterions and is the conceptual schema language which is most widely accepted in the world today.<br>INSPIRE requirement |
| | ISO 19109 [26] | Fulfills all criterions and is the conceptual schema language which is most widely accepted in the world today.<br>INSPIRE requirement |
| | | |
| Data content specification – Safety features | ISO 14825 – GDF [4]<br><br>INSPIRE Data specification on transport networks 0<br><br>EuroRoadS, deliverable D6.5 [35] | All of the referenced specifications contain potential definitions usable in the context of ROSATTE. |
| | | |
| Data content specification – Metadata definition | INSPIRE Metadata implementing rule [14]<br>ISO 19115 [23]<br>EuroRoadS D6.8 [32] | Fulfills all criterions |
| | | |
| Data content specification – Location referencing | ISO 17572-3, AGORA-C [3] | Fulfills all criterions.<br>The normal case for ROSATTE is that the database representations of the underlying road network differ between enacting authority and information provider. Therefore, an on-the-fly location referencing method is needed where available and significant attribution can be used to identify safety feature locations on the road network.<br><br>Note:<br>Currently, INSPIRE transport networks 0 only support linear referencing as location referencing method. This covers the case where linear referencing systems are shared between parties or when also the network ele- |

| | | |
|---|---|---|
| | | ments are exchanged. This method does not fulfill all the immediate requirements for ROSATTE. |
| | | |
| Data exchange specification – schema and physical format | ISO 19136 [27] | Fulfills all criterions. No specific requirements have been specified for ROSATTE regarding compactness and such which could have indicated the use of something more compact.<br>Recommended from INSPIRE |
| | | |
| Service specification | REST [34]<br>OGC Catalogue services [31] | Fulfills all criterions except #2.<br>SOAP/WSDL [29]/[30] is recommended by INSPIRE. However, since SOAP/WSDL is not a requirement and it is believed that a REST approach is better suited for ROSATTE purposes, REST is the current choice for ROSATTE. |

# 6. Data content specification

## 6.1. Introduction

To comply with INSPIRE and ISO 19100, a conceptual data content specification for ROSATTE has been developed according to ISO 19109 – Rules for application schema. The data content specification specifies the various classes of information that is being used within ROSATTE without mandating a particular technology for implementation. The data content specification uses UML constructs such as packages, classes, attributes, associations and OCL constraints and assumes that the reader is familiar in that domain.

The data content specification is organized in a couple of packages where each package corresponds to a separate subset of the ROSATTE domain. Each package contains a number of classes with attributes and associations.

Basically, the ROSATTE model defines the following basic concepts:

- **Datasets** consisting of safety features and metadata describing the content of the dataset. These datasets are supposed to be exchanged from enacting authorities to information providers (map providers). For the primary use-case, this type of dataset actually contains incremental updates for safety features that occurred in a ROSATTE data store since the last time data exchange took place. These updates make use of a unique ID identifier. It is the responsibility of the information-/map-provider to remember when the last successful dataset was created. The difference between a dataset containing incremental updates and a dataset containing a complete snapshot from a ROSATTE data store is explained in chapter 6.8
- Another type of dataset consisting of **feedback information** from information providers (map providers) to enacting authorities
- **Safety features** which describe the contents of traffic regulations. A safety feature is primarily described by:
    - Its type (speed limit, restriction for vehicles, overtaking ban etc)
    - Property values (speed limit value, maximum weight etc)
    - Conditions for when the safety feature is applicable (time period, vehicle type, weather conditions etc)
    - Location for explicit safety features. Implicit safety features normally has no explicit location
    - Update information. For datasets containing incremental updates, all safety features shall contain information on the applicable update primitive

## 6.2. Package overview

The figure below shows the different packages used within ROSATTE and their interdependencies.

The figure shows that a number of external definitions are being used:

- Regarding metadata, INSPIRE implementing rule is used and the concrete UML definitions are imported from ISO 19115 [23]. The exact definition in UML and text of the used metadata elements are not copied to this document
- Regarding location referencing, ISO 17572-3, AGORA-C is used almost as is. The few exceptions are listed in chapter 6.4. This chapter also defines the interface between ROSATTE and AGORA-C.
- Definitions from ISO 19103 [24], ISO 19108 [25], ISO 14825 and EuroRoadS (D6.5 [35]and D6.8 [32]) are being used whenever applicable

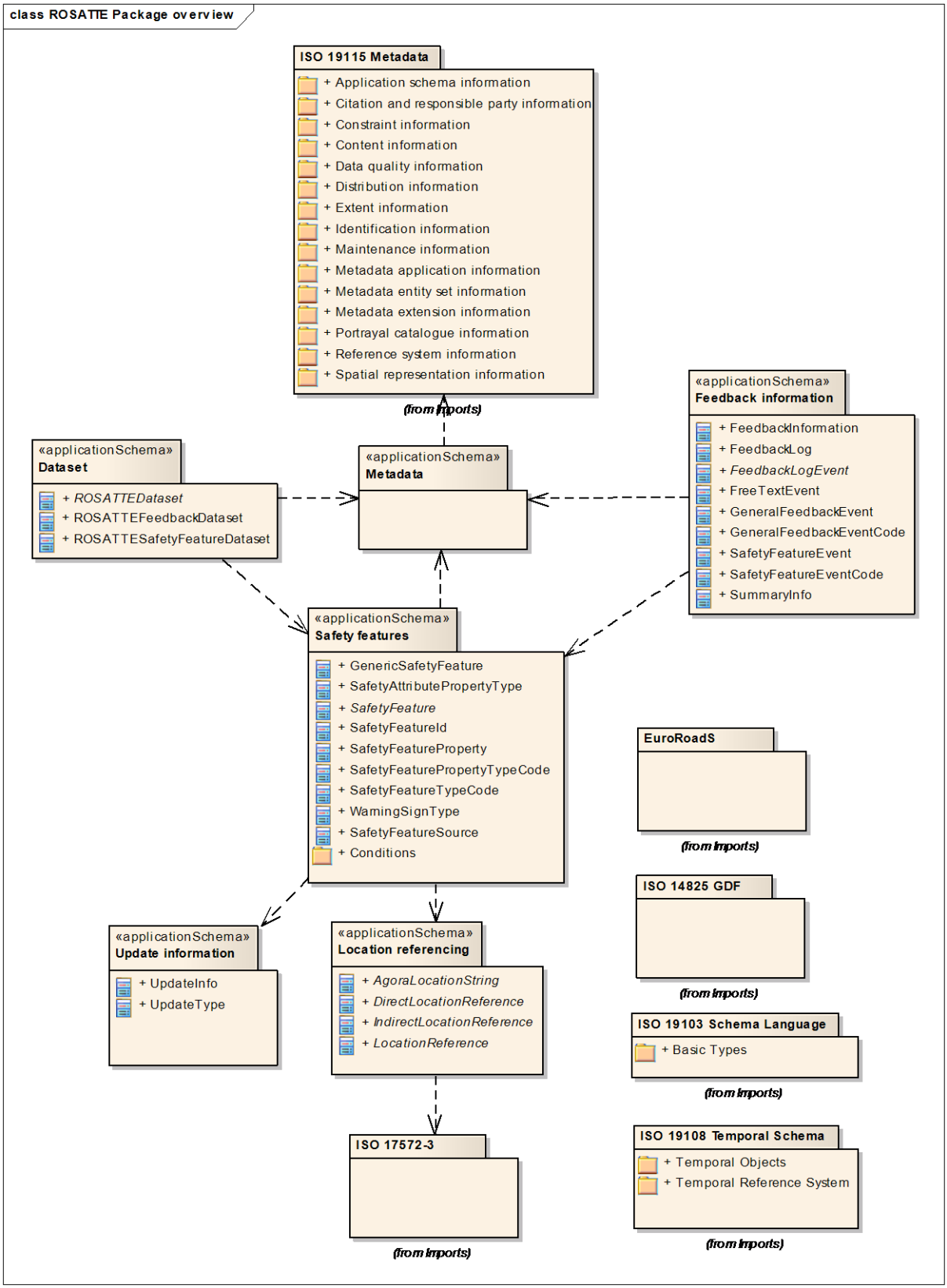



**Figure 4 - Package overview**

## 6.3. Package Safety features

This package defines the various safety features that may be used within ROSATTE. The package uses a generic and flexible modeling approach which means that the set of specific safety feature types and safety feature type properties may be extended by extending the appropriate codelists. This means that all safety feature types are handled by the same small set of UML classes. An alternative approach would have been to model each type of safety feature as a separate specific UML class. The main reasons for proposing the generic approach are the following:

+ Simple model with few classes
+ Easily extendable model
+ Simple to define the update service protocol

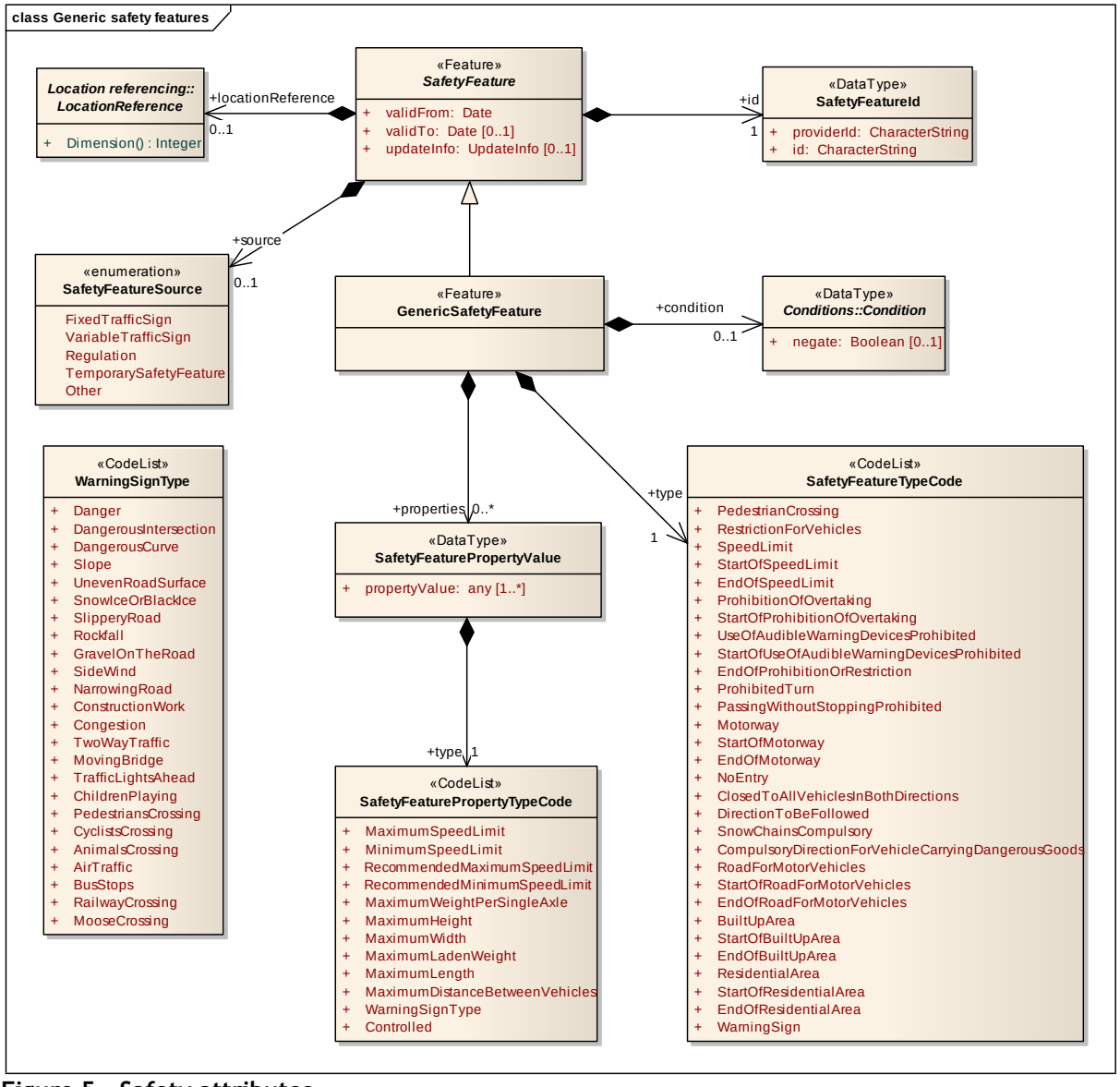



**Figure 5 - Safety attributes**

### 6.3.1. Class definitions

| Class<<Feature>>: SafetyFeature | | |
|---|---|---|
| | Definition: | An abstract class intended to serve as a base class for any type of safety feature. A safety feature represents the content of a traffic regulation in a digital road database. To be useful, each safety feature along a road must be paired by the description of its location. The location may be a point or a linear location |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | Feature |
| **Attribute: validFrom** | | |
| | Definition: | The date from which this safety feature is valid and in use |
| | Value type: | Date [ISO 19103] |
| | Multiplicity: | 1..1 |
| | Stereotypes: | |
| **Attribute: validTo** | | |
| | Definition: | The date from which this safety feature is no longer valid |
| | Value type: | Date [ISO 19103] |
| | Multiplicity | 0..1<br>Note: Value not present indicates that the safety feature is valid until further notice<br><br>Note: This value is typically set when the source of the safety feature is temporary |
| | Stereotypes: | |
| **Attribute: updateInfo** | | |
| | Definition: | Specifies the necessary update information if this is a representation of an update event. |
| | Value type: | UpdateInfo<br><br>Note:<br>This type is defined in the package **Update information** later in this document. The type defines the update primitives **"Add"**, **"Modify"** and **"Remove"** |
| | Multiplicity | 0..1<br>Note: Value not present indicates that the safety feature represents a state-of-the-art snapshot of existing data. |
| | Stereotypes: | |
| **Association role: source** | | |
| | Definition: | Specifies the source for the safety feature. |
| | Value type: | SafetyFeatureSource |
| | Multiplicity | 1 |
| | Stereotypes: | |
| | | |
| **Association role: id** | | |
| | Definition: | The identity for the safety feature. The identity shall follow the rules for permanent identification within ROSATTE. |
| | Value type: | SafetyAttributeId |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Association role: locationReference** | | |
| | Definition: | The location for the safety feature |
| | Value type: | LocationReference |
| | Multiplicity: | 0..1 |

| | | |
|---|---|---|
| | | Note:<br>The location reference may be left empty for implicit safety features where there is no exact location. An example of an implicit safety feature is a speed limit valid inside built up areas. |
| | Stereotypes: | |
| | | |

| **Class<<Feature>>: GenericSafetyFeature** | | |
|---|---|---|
| | Definition: | A generic (type independent) representation of safety feature |
| | Subtype of: | SafetyFeature |
| | Status: | Proposed |
| | Stereotypes: | <<Feature>> |
| **Association role: type** | | |
| | Definition: | The safety feature type |
| | Value type: | SafetyFeatureType |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Association role: properties** | | |
| | Definition: | The property values of interest for the safety feature |
| | Value type: | SafetyFeatureProperty |
| | Multiplicity | 0..* |
| | Stereotypes: | |
| **Association role: condition** | | |
| | Definition: | Describes the circumstances when the safety feature is applicable. |
| | Value type: | SafetyFeatureCondition |
| | Multiplicity | 0..1<br>Note: If empty, the safety feature is unconditional. |
| | Stereotypes: | |
| **Constraint: PedestrianCrossing**<br>Natural language:<br>Safety feature type PedestrianCrossing requires the set of properties to be empty or that the property type Controlled is used<br>OCL:<br>type = **PedestrianCrossing** implies<br>properties->isEmpty() or (properties->size() = 1 and properties->forall(p \| p.type = **Controlled**)) | | |
| **Constraint: RestrictionForVehicles**<br>Natural language:<br>Safety feature type RestrictionForVehicles requires one of property types MaximumWeightPerSingleAxle, MaximumHeight, MaximumWidth, MaximumLadenWeight, MaximumLength or MaximumDistanceBetweenVehicles to be used<br>OCL:<br>type = **RestrictionForVehicles** implies<br>properties->forall(p \| p.type = **MaximumWeightPerSingleAxle** or p.type = **MaximumHeight** or p.type = **MaximumWidth** or p.type = **MaximumLadenWeight** or p.type = **MaximumLength** or p.type = **MaximumDistanceBetweenVehicles)** | | |
| **Constraint: SpeedLimit**<br>Natural language:<br>Safety feature type SpeedLimit requires one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **SpeedLimit** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() = 1 | | |
| **Constraint: SpeedLimitLocation** | | |

| **Class<<Feature>>: GenericSafetyFeature** |
|---|
| Natural language:<br>Safety feature type SpeedLimit requires a linear location to be used<br>OCL:<br>type = **SpeedLimit** implies<br>locationReference->size() = 0 or (locationReference.size() = 1 and locationReference.Dimension() = 1) |
| **Constraint: StartOfSpeedLimit**<br>Natural language:<br>Safety feature type StartOfSpeedLimit requires one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **StartOfSpeedLimit** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() = 1 |
| **Constraint: StartOfSpeedLimitLocation**<br>Natural language:<br>Safety feature type StartOfSpeedLimit requires a point location to be used<br>OCL:<br>type = **StartOfSpeedLimit** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: EndOfSpeedLimit**<br>Natural language:<br>Safety feature type EndOfSpeedLimit requires one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **EndOfSpeedLimit** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() = 1 |
| **Constraint: EndOfSpeedLimitLocation**<br>Natural language:<br>Safety feature type EndOfSpeedLimit requires a point location to be used<br>OCL:<br>type = **EndOfSpeedLimit** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: ProhibitionOfOvertaking**<br>Natural language:<br>Safety feature type ProhibitionOfOvertaking requires the set of properties to be empty<br>OCL:<br>type = **ProhibitionOfOvertaking** implies<br>properties->isEmpty() |
| **Constraint: ProhibitionOfOvertakingLocation**<br>Natural language:<br>Safety feature type ProhibitionOfOvertaking requires a linear location to be used<br>OCL:<br>type = **ProhibitionOfOvertakingLocation** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: StartOfProhibitionOfOvertaking**<br>Natural language:<br>Safety feature type StartOfProhibitionOfOvertaking requires the set of properties to be empty<br>OCL:<br>type = **StartOfProhibitionOfOvertaking** implies |

| **Class<<Feature>>: GenericSafetyFeature** |
|---|
| properties->isEmpty() |
| **Constraint: StartOfProhibitionOfOvertakingLocation**<br>Natural language:<br>Safety feature type StartOfProhibitionOfOvertaking requires a point location to be used<br>OCL:<br>type = **StartOfProhibitionOfOvertaking** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: UseOfAudibleWarningDevicesProhibited**<br>Natural language:<br>Safety feature type UseOfAudibleWarningDevicesProhibited requires the set of properties to be empty<br>OCL:<br>type = **UseOfAudibleWarningDevicesProhibited** implies<br>properties->isEmpty() |
| **Constraint: StartOfUseOfAudibleWarningDevicesProhibited**<br>Natural language:<br>Safety feature type StartOfUseOfAudibleDevicesProhibited requires a linear location to be used<br>OCL:<br>type = **StartOfUseOfAudibleWarningDevicesProhibited** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: EndOfProhibitionOrRestriction**<br>Natural language:<br>Safety feature type EndOfProhibitionOrRestriction requires the set of properties to be empty<br>OCL:<br>type = **EndOfProhibitionOrRestriction** implies<br>properties->isEmpty() |
| **Constraint: EndOfProhibitionOrRestrictionLocation**<br>Natural language:<br>Safety feature type EndOfProhibitionOrRestriction requires a point location to be used<br>OCL:<br>type = **EndOfProhibitionOrRestriction** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: ProhibitedTurn**<br>Natural language:<br>Safety feature type ProhibitedTurn requires the set of properties to be empty<br>OCL:<br>type = **ProhibitedTurn** implies<br>properties->isEmpty() |
| **Constraint: PassingWithoutStoppingProhibited**<br>Natural language:<br>Safety feature type PassingWithoutStoppingProhibited requires the set of properties to be empty<br>OCL:<br>type = **PassingWithoutStoppingProhibited** implies<br>properties->isEmpty() |
| **Constraint: Motorway**<br>Natural language:<br>Safety feature type Motorway allows one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **Motorway** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() <= 1 |
| **Constraint: MotorwayLocation**<br>Natural language: |

| **Class<<Feature>>: GenericSafetyFeature** |
|---|
| Safety feature type motorway requires a linear location to be used<br>OCL:<br>type = **Motorway** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: StartOfMotorway**<br>Natural language:<br>Safety feature type StartOfMotorway allows one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **StartOfMotorway** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() <= 1 |
| **Constraint: StartOfMotorwayLocation**<br>Natural language:<br>Safety feature type StartOfMotorway requires a point location to be used<br>OCL:<br>type = **StartOfMotorway** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: EndOfMotorway**<br>Natural language:<br>Safety feature type EndOfMotorway requires the set of properties to be empty<br>OCL:<br>type = **EndOfMotorway** implies<br>properties->isEmpty() |
| **Constraint: EndOfMotorwayLocation**<br>Natural language:<br>Safety feature type EndOfMotorway requires a point location to be used<br>OCL:<br>type = **EndOfMotorway** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: NoEntry**<br>Natural language:<br>Safety feature type NoEntry requires the set of properties to be empty<br>OCL:<br>type = **NoEntry** implies<br>properties->isEmpty() |
| **Constraint: ClosedToAllVehiclesInBothDirections**<br>Natural language:<br>Safety feature type ClosedToAllVehiclesInBothDirections requires the set of properties to be empty<br>OCL:<br>type = **ClosedToAllVehiclesInBothDirections** implies<br>properties->isEmpty() |
| **Constraint: DirectionToBeFollowed**<br>Natural language:<br>Safety feature type DirectionToBeFollowed requires the set of properties to be empty<br>OCL:<br>type = **DirectionToBeFollowed** implies<br>properties->isEmpty() |
| **Constraint: SnowChainsCompulsory**<br>Natural language:<br>Safety feature type SnowChainsCompulsory requires the set of properties to be empty<br>OCL:<br>type = **SnowChainsCompulsory** implies<br>properties->isEmpty() |

| Class<<Feature>>: GenericSafetyFeature |
|---|
| **Constraint: CompulsoryDirectionForVehicleCarryingDangerousGoods**<br>Natural language:<br>Safety feature type CompulsoryDirectionForVehicleCarryingDangerousGoods requires the set of properties to be empty<br>OCL:<br>type = **CompulsoryDirectionForVehicleCarryingDangerousGoods** implies<br>properties->isEmpty() |
| **Constraint: RoadForMotorVehicles**<br>Natural language:<br>Safety feature type RoadForMotorVehicles requires the set of properties to be empty<br>OCL:<br>type = **RoadForMotorVehicles** implies<br>properties->isEmpty() |
| **Constraint: RoadForMotorVehiclesLocation**<br>Natural language:<br>Safety feature type RoadForMotorVehicles requires a linear location to be used<br>OCL:<br>type = **RoadForMotorVehicles** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: StartOfRoadForMotorVehicles**<br>Natural language:<br>Safety feature type StartOfRoadForMotorVehicles requires the set of properties to be empty<br>OCL:<br>type = **StartOfRoadForMotorVehicles** implies<br>properties->isEmpty() |
| **Constraint: StartOfRoadForMotorVehiclesLocation**<br>Natural language:<br>Safety feature type StartOfRoadForMotorVehicles requires a point location to be used<br>OCL:<br>type = **StartOfRoadForMotorVehicles** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: EndOfRoadForMotorVehicles**<br>Natural language:<br>Safety feature type EndOfRoadForMotorVehicles requires the set of properties to be empty<br>OCL:<br>type = **EndOfRoadForMotorVehicles** implies<br>properties->isEmpty() |
| **Constraint: EndOfRoadForMotorVehiclesLocation**<br>Natural language:<br>Safety feature type EndOfRoadForMotorVehicles requires a point location to be used<br>OCL:<br>type = **EndOfRoadForMotorVehicles** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: BuiltUpArea**<br>Natural language:<br>Safety feature type BuiltUpArea allows one of property types MaximumSpeedLimit, MinimumSpeed-Limit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **BuiltUpArea** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and proper-ties->size() <= 1 |
| **Constraint: BuiltUpAreaLocation**<br>Natural language:<br>Safety feature type BuiltUpArea requires a linear location to be used |

| **Class<<Feature>>: GenericSafetyFeature** |
|---|
| OCL:<br>type = **BuiltUpArea** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: StartOfBuiltUpArea**<br>Natural language:<br>Safety feature type StartOfBuiltUpArea allows one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **StartOfBuiltUpArea** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() <= 1 |
| **Constraint: StartOfBuiltUpAreaLocation**<br>Natural language:<br>Safety feature type StartOfBuiltUpArea requires a point location to be used<br>OCL:<br>type = **StartOfBuiltUpArea** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: EndOfBuiltUpArea**<br>Natural language:<br>Safety feature type EndOfBuiltUpArea requires the set of properties to be empty<br>OCL:<br>type = **EndOfBuiltUpArea** implies<br>properties->isEmpty() |
| **Constraint: EndOfBuiltUpAreaLocation**<br>Natural language:<br>Safety feature type EndOfBuiltUpArea requires a point location to be used<br>OCL:<br>type = **EndOfBuiltUpArea** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: ResidentialArea**<br>Natural language:<br>Safety feature type ResidentialArea allows one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **ResidentialArea** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and properties->size() <= 1 |
| **Constraint: ResidentialAreaLocation**<br>Natural language:<br>Safety feature type ResidentialArea requires a linear location to be used<br>OCL:<br>type = **ResidentialArea** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 1 |
| **Constraint: StartOfResidentialArea**<br>Natural language:<br>Safety feature type StartOfResidentialArea allows one of property types MaximumSpeedLimit, MinimumSpeedLimit, RecommendedMaximumSpeedLimit or RecommendedMinimumSpeedLimit to be used<br>OCL:<br>type = **StartOfResidentialArea** implies<br>properties->forall(p \| p.type = **MaximumSpeedLimit** or p.type = **MinimumSpeedLimit** or p.type = **RecommendedMaximumSpeedLimit** or p.type = **RecommendedMinimumSpeedLimit)** and proper- |

| **Class<<Feature>>: GenericSafetyFeature** |
|---|
| ties->size() <= 1 |
| **Constraint: StartOfResidentialAreaLocation**<br>Natural language:<br>Safety feature type StartOfResidentialArea requires a point location to be used<br>OCL:<br>type = **StartOfResidentialArea** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: EndOfResidentialArea**<br>Natural language:<br>Safety feature type EndOfResidentialArea requires the set of properties to be empty<br>OCL:<br>type = **EndOfResidentialArea** implies<br>properties->isEmpty() |
| **Constraint: EndOfResidentialAreaLocation**<br>Natural language:<br>Safety feature type EndOfResidentialArea requires a point location to be used<br>OCL:<br>type = **EndOfResidentialArea** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |
| **Constraint: WarningSign**<br>Natural language:<br>Safety feature type WarningSign requires the property type WarningSignType to be used and allows the property type Controlled to be used<br>OCL:<br>type = **WarningSign** implies<br>(properties->forall(p \| p.type = **WarningSignType** or p.type = **Controlled)** and properties->size() >= 1) and (properties->select(type = **WarningSignType**)->size() = 1) |
| **Constraint: WarningSignLocation**<br>Natural language:<br>Safety feature type WarningSign requires a point location to be used<br>OCL:<br>type = **WarningSign** implies<br>locationReference.size() = 1 and locationReference.Dimension() = 0 |

| **Class<<DataType>>: SafetyFeatureProperty** | | |
|---|---|---|
| | Definition: | A generic (type independent) representation of a safety feature property value |
| | Subtype of: | |
| | Status: | Proposed |
| | Stereotypes: | DataType |
| **Attribute: propertyValue** | | |
| | Definition: | The value of the safety feature property |
| | Value type: | Any<br>Note: The actual value type is dependent on type |
| | Multiplicity: | 1..* |
| | Stereotypes: | |
| **Association role: type** | | |
| | Definition: | The type of safety attribute property |
| | Value type: | SafetyFeaturePropertyTypeCode |
| | Multiplicity | 1 |
| | Stereotypes: | |
| **Constraint: SafetyFeaturePropertyTypes**<br>Natural language:<br>Each safety feature property type implies a certain type of property value | | |

| Class<<DataType>>: SafetyFeatureProperty |
|---|
| OCL:<br>type = **MaximumSpeedLimit** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Velocity**))<br>type = **MinimumSpeedLimit** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Velocity**))<br>type = **RecommendedMaximumSpeedLimit** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Velocity**))<br>type = **RecommendedMinimumSpeedLimit** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Velocity**))<br><br>type = **MaximumWeightPerSingleAxle** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Weight**))<br>type = **MaximumHeight** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Length**))<br>type = **MaximumWidth** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Length**))<br>type = **MaximumLadenWeight** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Weigth**))<br>type = **MaximumLength** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Length**))<br>type = **MaximumDistanceBetweenVehicles** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Length**))<br>type = **WarningSignType** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**WarningSignType**)<br>type = **Controlled** implies<br>propertyValue->forall(v \| v.oclIsKindOf(**Boolean**)<br><br>Note: Velocity, Weight, Length and Distance are defined in ISO 19103 [24] |

| Class <<CodeList>>: SafetyFeatureTypeCode | | |
|---|---|---|
| | Definition: | Specifies the various safety feature type codes. |
| | Status: | Proposed |
| | Stereotypes: | CodeList |
| **Value: PedestrianCrossing** | | |
| | Definition: | Pedestrian crossing |
| | Code: | 1 |
| **Value: RestrictionForVehicles** | | |
| | Definition: | Restriction for vehicles regarding weight, height or length |
| | Code: | 2 |
| **Value: SpeedLimit** | | |
| | Definition: | Speed limit |
| | Code: | 3 |
| **Value: StartOfSpeedLimit** | | |
| | Definition: | Start of speed limit |
| | Code: | 4 |
| **Value: EndOfSpeedLimit** | | |
| | Definition: | End of speed limit |
| | Code: | 5 |
| **Value: ProhibitionOfOvertaking** | | |
| | Definition: | Prohibition of overtaking |
| | Code: | 6 |
| **Value: StartOfProhibitionOfOvertaking** | | |
| | Definition: | Start of prohibition of overtaking |

| Class <<CodeList>>: SafetyFeatureTypeCode | | |
|---|---|---|
| | Code: | 7 |
| **Value: UseOfAudibleWarningDevicesProhibited** | | |
| | Definition: | Use of audible warning devices prohibited |
| | Code: | 8 |
| **Value: StartOfUseOfAudibleWarningDevicesProhibited** | | |
| | Definition: | Start of use of audible warning devices prohibited |
| | Code: | 9 |
| **Value: EndOfProhibitionOrRestriction** | | |
| | Definition: | End of prohibition or restriction. Linked to 2-9 and means "end of all local prohibitions". For speed limits the generic speed limits apply again |
| | Code: | 10 |
| **Value: ProhibitedTurn** | | |
| | Definition: | Prohibited turn |
| | Code: | 11 |
| **Value: PassingWithoutStoppingProhibited** | | |
| | Definition: | Passing without stopping prohibited. Applicable for customs, police, toll-booths |
| | Code: | 12 |
| **Value: Motorway** | | |
| | Definition: | Motorway. Generally implies a different generic speed limit. |
| | Code: | 13 |
| **Value: StartOfMotorway** | | |
| | Definition: | StartOfMotorway. Generally implies a different generic speed limit. |
| | Code: | 14 |
| **Value: EndOfMotorway** | | |
| | Definition: | End of motorway. Generally implies a different generic speed limit. |
| | Code: | 15 |
| **Value: NoEntry** | | |
| | Definition: | No entry |
| | Code: | 16 |
| **Value: ClosedToAllVehiclesInBothDirections** | | |
| | Definition: | Closed to all vehicles in both directions |
| | Code: | 17 |
| **Value: DirectionToBeFollowed** | | |
| | Definition: | Direction to be followed |
| | Code: | 18 |
| **Value: SnowChainsCompulsory** | | |
| | Definition: | Snow chains compulsory |
| | Code: | 19 |
| **Value: CompulsoryDirectionForVehicleCarryingDangerousGoods** | | |
| | Definition: | Compulsory direction for vehicles carrying dangerous goods |
| | Code: | 20 |
| **Value: RoadForMotorVehicles** | | |
| | Definition: | Road for motor vehicles |
| | Code: | 21 |
| **Value: StartOfRoadForMotorVehicles** | | |
| | Definition: | Start of road for motor vehicles |
| | Code: | 22 |
| **Value: EndOfRoadForMotorVehicles** | | |
| | Definition: | End of road for motor vehicles |
| | Code: | 23 |
| **Value: BuiltUpArea** | | |
| | Definition: | Built up area. Implies a different generic speed limit |

| Class <<CodeList>>: SafetyFeatureTypeCode | | |
|---|---|---|
| | Code: | 24 |
| **Value: StartOfBuiltUpArea** | | |
| | Definition: | Start of built up area. Implies a different generic speed limit |
| | Code: | 25 |
| **Value: EndOfBuiltUpArea** | | |
| | Definition: | End of built up area. Implies a different generic speed limit |
| | Code: | 26 |
| **Value: ResidentialArea** | | |
| | Definition: | Residential area. Implies a dedicated generic speed limit |
| | Code: | 27 |
| **Value: StartOfResidentialArea** | | |
| | Definition: | Residential area. Implies a dedicated generic speed limit |
| | Code: | 28 |
| **Value: EndOfResidentialArea** | | |
| | Definition: | End of residential area. Implies a generic speed limit |
| | Code: | 29 |
| **Value: WarningSign** | | |
| | Definition: | Warning sign. |
| | Code: | 30 |

| Class <<CodeList>>: SafetyFeaturePropertyTypeCode | | |
|---|---|---|
| | Definition: | Specifies the various safety attribute property type codes |
| | Status: | Proposed |
| | Stereotypes: | CodeList |
| **Value: MaximumSpeed** | | |
| | Definition: | Maximum speed |
| | Code: | 1 |
| **Value: MinimumSpeed** | | |
| | Definition: | Minimum speed |
| | Code: | 2 |
| **Value: RecommendedMaximumSpeed** | | |
| | Definition: | Recommended maximum speed |
| | Code: | 3 |
| **Value: RecommendedMinimumSpeed** | | |
| | Definition: | Recommended minimum speed |
| | Code: | 4 |
| **Value: MaximumWeightPerSingleAxle** | | |
| | Definition: | Maximum weight per single axle |
| | Code: | 3 |
| **Value: MaximumHeight** | | |
| | Definition: | Maximum height |
| | Code: | 4 |
| **Value: MaximumWidth** | | |
| | Definition: | Maximum width |
| | Code: | 5 |
| **Value: MaximumLadenWeight** | | |
| | Definition: | Maximum laden weight |
| | Code: | 6 |
| **Value: MaximumLength** | | |
| | Definition: | Maximum length |
| | Code: | 7 |
| **Value: MaximumDistanceBetweenVehicles** | | |
| | Definition: | Maximum distance between vehicles |

| Class <<CodeList>>: SafetyFeaturePropertyTypeCode | | |
|---|---|---|
| | Code: | 8 |
| **Value: WarningSignType** | | |
| | Definition: | Warning sign type |
| | Code: | 9 |
| **Value: Controlled** | | |
| | Definition: | Indication of whether a pedestrian crossing is controlled (by signalling) or not |
| | Code: | 10 |

| Class <<enumeration>>: SafetyFeatureSource | | |
|---|---|---|
| | Definition: | Specifies the possible sources for a safety feature |
| | Status: | Proposed |
| | Stereotypes: | Enumeration |
| **Value: FixedTrafficSign** | | |
| | Definition: | The source is a fixed traffic sign |
| | Code: | 1 |
| **Value: VariableTrafficSign** | | |
| | Definition: | The source is a variable traffic sign |
| | Code: | 2 |
| **Value: Regulation** | | |
| | Definition: | The source is a legal regulation |
| | Code: | 3 |
| **Value: TemporarySafetyFeature** | | |
| | Definition: | The source is a temporary safety feature.<br><br>Note:<br>Regulation indicating a speed limit that is in force during a predefined time period. The speed limit can also vary during this time period. For example, road works, or a speed reduction in a school zone during school time. The time during which the speed limit is in force is also signposted." (reference: SpeedAlert Consortium, "D2.1 Common Definition of Speed Limits and Classifications", (final) version 1.5, November 2004) |
| | Code: | 4 |
| **Value: Other** | | |
| | Definition: | Other source |
| | Code: | 5 |

| Class <<CodeList>>: WarningSignType | | |
|---|---|---|
| | Definition: | Specifies the various types of warning signs |
| | Status: | Proposed |
| | Stereotypes: | CodeList |
| **Value: Danger** | | |
| | Definition: | Danger |
| | Code: | 1 |
| **Value: DangerousIntersection** | | |
| | Definition: | Dangerous intersection |
| | Code: | 2 |
| **Value: DangerousCurve** | | |
| | Definition: | Dangerous curve |
| | Code: | 3 |
| **Value: Slope** | | |
| | Definition: | Slope |
| | Code: | 4 |

| Class <<CodeList>>: WarningSignType | | |
|---|---|---|
| **Value: UnevenRoadSurface** | | |
| | Definition: | Un-even road surface |
| | Code: | 5 |
| **Value: SnowIceOrBlackIce** | | |
| | Definition: | Snow, ice or black ice |
| | Code: | 6 |
| **Value: SlipperyRoad** | | |
| | Definition: | Slippery road |
| | Code: | 7 |
| **Value: Rockfall** | | |
| | Definition: | Rockfall |
| | Code: | 8 |
| **Value: GravelOnTheRoad** | | |
| | Definition: | Gravel on the road |
| | Code: | 9 |
| **Value: SideWind** | | |
| | Definition: | Sidewind |
| | Code: | 10 |
| **Value: NarrowingRoad** | | |
| | Definition: | Narrowing road |
| | Code: | 11 |
| **Value: ConstructionWork** | | |
| | Definition: | Construction work |
| | Code: | 12 |
| **Value: Congestion** | | |
| | Definition: | Congestion |
| | Code: | 13 |
| **Value: TwoWayTraffic** | | |
| | Definition: | Two way traffic (when there was one-way before) |
| | Code: | 14 |
| **Value: MovingBridge** | | |
| | Definition: | Moving bridge |
| | Code: | 15 |
| **Value: TrafficLightsAhead** | | |
| | Definition: | Traffic lights ahead |
| | Code: | 16 |
| **Value: ChildrenPlaying** | | |
| | Definition: | Children playing |
| | Code: | 17 |
| **Value: PedestriansCrossing** | | |
| | Definition: | Pedestrians crossing<br>Note:<br>This warning sign type indicates the existence of a warning sign for pede-strians crossing. The SafetyFeatureType PedestrianCrossing indicates the crossing itself. |
| | Code: | 18 |
| **Value: CyclistsCrossing** | | |
| | Definition: | Cyclists crossing |
| | Code: | 19 |
| **Value: AnimalsCrossing** | | |
| | Definition: | Animals crossing (dear or cow on the sign) |
| | Code: | 20 |
| **Value: AirTraffic** | | |

| Class <<CodeList>>: WarningSignType | | |
|---|---|---|
| | Definition: | Air traffic |
| | Code: | 21 |
| **Value: BusStops** | | |
| | Definition: | Bus stops |
| | Code: | 22 |
| **Value: RailwayCrossing** | | |
| | Definition: | Railway crossing |
| | Code: | 23 |
| **Value: MooseCrossing** | | |
| | Definition: | Moose crossing |
| | Code: | 24 |

### 6.3.2. *Sub package Conditions*

This package defines the classes necessary to be able to represent safety feature conditions. These conditions represent situations for which the safety feature is applicable. For example can a speed limit be applicable only during a certain time period or for certain vehicles.

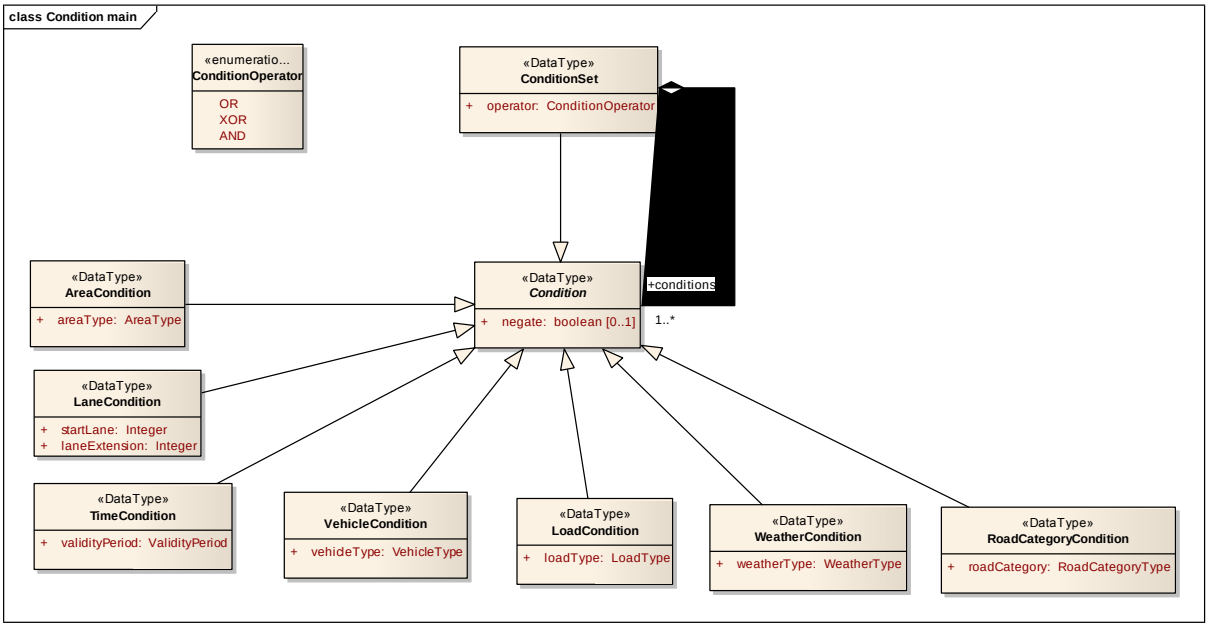



**Figure 6 – Safety feature conditions overview**

| Class<<DataType>>: Condition | | |
|---|---|---|
| | Definition: | Abstract class that specifies a condition for the validity of a certain safety feature such as time, weather conditions, vehicle etc.<br>The data structure allows for conditions such as:<br><ConditionSet operator="OR"><br><ConditionSet operator="AND"><br><TimeCondition…/><br><VehicleCondition…/><br></ConditionSet><br><ConditionSet operator="AND"><br><TimeCondition…/><br><VehicleCondition…/><br></ConditionSet> |

| Class<<DataType>>: Condition | | |
|---|---|---|
| | | </ConditionSet><br>Or, the same example in an alternate notation:<br>((TimeCondition and VehicleCondition) or (TimeCondition and VehicleCondition)) |
| | Subtype of: | |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: negate** | | |
| | Definition: | If set to true means that the entire condition shall be negated.<br>Note:<br>A condition stating VehicleType=EmergencyVehicle and negate = true would mean all vehicles except those of type EmergencyVehicle. |
| | Value type: | Boolean |
| | Multiplicity: | 0..1<br>Note:<br>Unset value means the same as false, i.e. no negation applied. |
| | Stereotypes: | |
| | | |

| Class<<DataType>>: ConditionSet | | |
|---|---|---|
| | Definition: | Groups a number of conditions in a set and applying a Boolean operator between the conditions in the set. |
| | Subtype of: | Condition |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: operator** | | |
| | Definition: | Specifies the operator to be applied between the conditions in the set |
| | Value type: | ConditionOperator |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Association role: conditions** | | |
| | Definition: | Specifies the conditions that are present in the set |
| | Value type: | Condition |
| | Multiplicity: | 1..* |
| | Stereotypes: | |

| Class <<DataType>>: LaneCondition | | |
|---|---|---|
| | Definition: | Specifies a condition regarding lanes |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: startLane** | | |
| | Definition: | Start lane counted from the right side for which the condition applies.<br><br>Note:<br>“Right side” is to be interpreted in relation to the direction of the location reference. |
| | Value type: | Integer |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: laneExtension** | | |
| | Definition: | Number of lanes including the start lane counted from the right hand side for which the condition applies. |
| | Value type: | Integer |

| Class <<DataType>>: LaneCondition | | |
|---|---|---|
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class<<DataType>>: TimeCondition | | |
|---|---|---|
| | Definition: | Specifies a condition concerning time |
| | Subtype of: | Condition |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: validityPeriod** | | |
| | Definition: | Specifies the time period for which this time condition applies |
| | Value type: | ValidityPeriod |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class<<DataType>>: VehicleCondition | | |
|---|---|---|
| | Definition: | Specifies a condition concerning vehicle type |
| | Subtype of: | Condition |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: vehicleType** | | |
| | Definition: | Specifies the vehicle type for which this condition applies |
| | Value type: | VehicleType |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class<<DataType>>: LoadCondition | | |
|---|---|---|
| | Definition: | Specifies a condition concerning load type |
| | Subtype of: | Condition |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: loadType** | | |
| | Definition: | Specifies the load type for which this condition applies |
| | Value type: | LoadType |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class<<DataType>>: WeatherCondition | | |
|---|---|---|
| | Definition: | Specifies a condition concerning weather type |
| | Subtype of: | Condition |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: weatherType** | | |
| | Definition: | Specifies the weather type for which this condition applies |
| | Value type: | WeatherType |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class<<DataType>>: AreaCondition | | |
|---|---|---|
| | Definition: | Specifies a condition concerning area type |
| | Subtype of: | Condition |
| | Status: | Proposed |

| Class<<DataType>>: AreaCondition | | |
|---|---|---|
| | Stereotypes: | <<DataType>> |
| **Attribute: areaType** | | |
| | Definition: | Specifies the area type for which this condition applies |
| | Value type: | AreaType |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class<<DataType>>: RoadCategoryCondition | | |
|---|---|---|
| | Definition: | Specifies a condition concerning road category<br>Note: This type of condition is primarily used when the actual safety feature has no specific location but specifies an implicit safety feature such as an implicit speed limit on motorways |
| | Subtype of: | Condition |
| | Status: | Proposed |
| | Stereotypes: | <<DataType>> |
| **Attribute: roadCategory** | | |
| | Definition: | Specifies the time period for which this time condition applies |
| | Value type: | RoadCategoryType |
| | Multiplicity: | 1 |
| | Stereotypes: | |

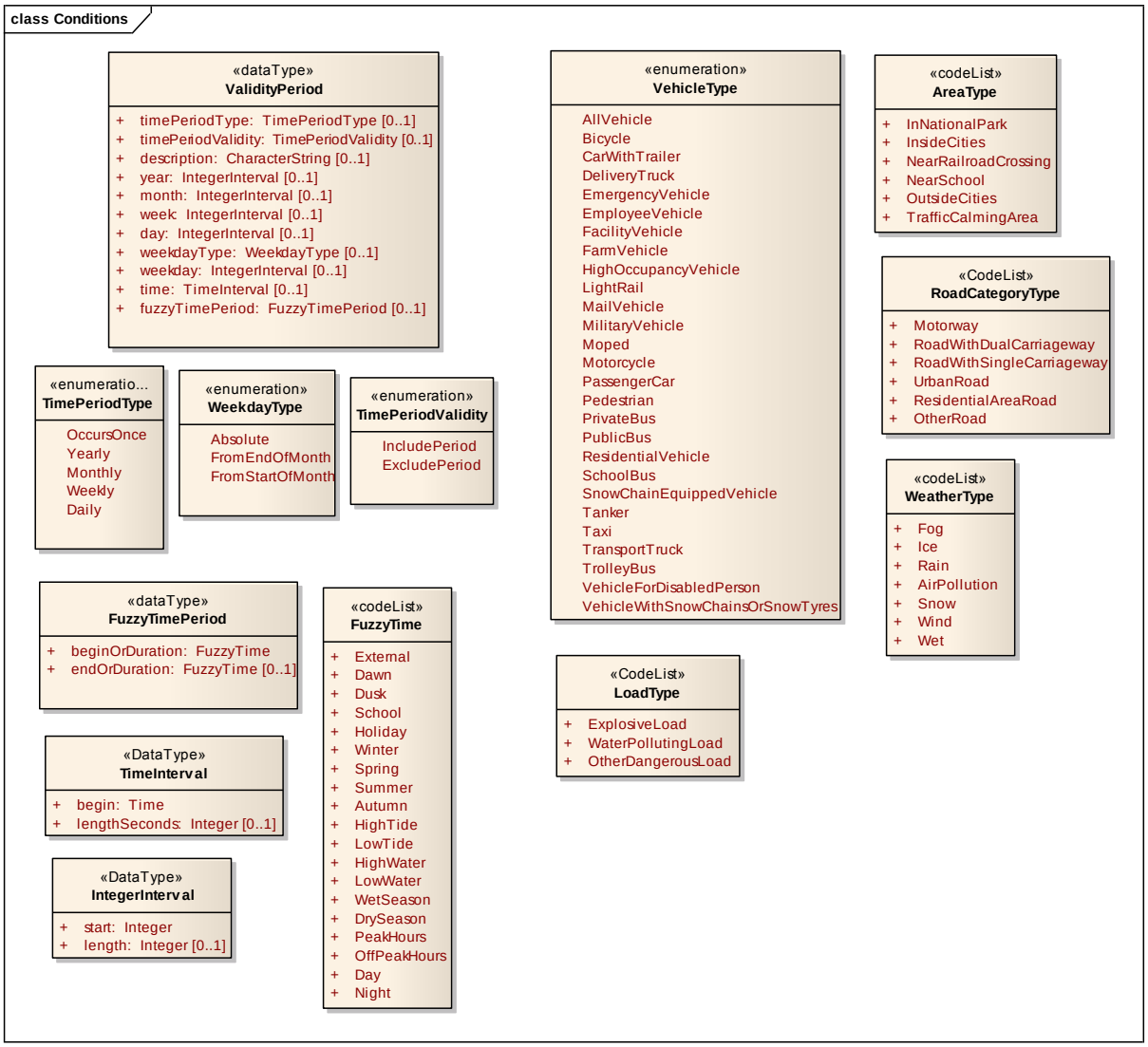



**Figure 7 - Safety feature conditions**

| Class <<CodeList>>: AreaType | | |
|---|---|---|
| | Definition: | Specifies the possible area types |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: InNationalPark** | | |
| | Definition: | In national park |
| | Code: | 1 |
| **Value: InsideCities** | | |
| | Definition: | Inside cities |
| | Code: | 2 |
| **Value: NearRailroadCrossing** | | |
| | Definition: | Near railroad crossing |
| | Code: | 3 |
| **Value: NearSchool** | | |
| | Definition: | Near school |
| | Code: | 4 |
| **Value: OutsideCities** | | |

| Class <<CodeList>>: AreaType | | |
|---|---|---|
| | Definition: | Outside cities |
| | Code: | 5 |
| **Value: TrafficCalmingArea** | | |
| | Definition: | Traffic calming area |
| | Code: | 6 |

| Class <<DataType>>: ValidityPeriod | | |
|---|---|---|
| | Definition: | Specifies the possible validity period. |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: timePeriodType** | | |
| | Definition: | Specifies if the time period occurs once or are repeated |
| | Value type: | TimePeriodType |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: timePeriodValidity** | | |
| | Definition: | Specifies if the time period is included or excluded |
| | Value type: | TimePeriodValidity |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: description** | | |
| | Definition: | Describes the time period |
| | Value type: | CharacterString |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: year** | | |
| | Definition: | Indicates the starting year and duration |
| | Value type: | IntegerInterval |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: month** | | |
| | Definition: | Indicates the starting month and duration |
| | Value type: | IntegerInterval |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: week** | | |
| | Definition: | Indicates the starting week and duration |
| | Value type: | IntegerInterval |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: day** | | |
| | Definition: | Indicates the starting day and duration |
| | Value type: | IntegerInterval |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: weekdayType** | | |
| | Definition: | Defines from where the weekday is counted |
| | Value type: | WeekdayType |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: weekday** | | |
| | Definition: | Indicates the starting weekday and duration |
| | Value type: | IntegerInterval |

| Class <<DataType>>: ValidityPeriod | | |
|---|---|---|
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: time** | | |
| | Definition: | Indicates the starting time and duration |
| | Value type: | TimeInterval |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: fuzzyTimePeriod** | | |
| | Definition: | Indicates the fuzzy time period |
| | Value type: | FuzzyTimePeriod |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |

| Class <<CodeList>>: WeatherType | | |
|---|---|---|
| | Definition: | Specifies the possible weather types |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: Fog** | | |
| | Definition: | Fog |
| | Code: | 1 |
| **Value: Ice** | | |
| | Definition: | Ice |
| | Code: | 2 |
| **Value: Rain** | | |
| | Definition: | Rain |
| | Code: | 3 |
| **Value: AirPollution** | | |
| | Definition: | Air pollution |
| | Code: | 4 |
| **Value: Snow** | | |
| | Definition: | Snow |
| | Code: | 5 |
| **Value: Wind** | | |
| | Definition: | Wind |
| | Code: | 6 |
| **Value: Wet** | | |
| | Definition: | Wet |
| | Code: | 7 |

| Class <<CodeList>>: VehicleType | | |
|---|---|---|
| | Definition: | Specifies the possible vehicle types |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: AllVehicle** | | |
| | Definition: | Any vehicle, not including pedestrians |
| | Code: | 1 |
| **Value: Bicycle** | | |
| | Definition: | Pedal-driven two-wheeled vehicle |
| | Code: | 2 |
| **Value: CarWithTrailer** | | |
| | Definition: | A passenger car with an attached trailer |
| | Code: | 3 |

| Class <<CodeList>>: VehicleType | | |
|---|---|---|
| **Value: DeliveryTruck** | | |
| | Definition: | A truck vehicle of relatively small size, whose principal use is for delivery of goods and materials |
| | Code: | 4 |
| **Value: CarWithTrailer** | | |
| | Definition: | A passenger car with an attached trailer |
| | Code: | 5 |
| **Value: EmergencyVehicle** | | |
| | Definition: | A vehicle engaged in emergency response, included but not limited to police, ambulance and fire |
| | Code: | 6 |
| **Value: EmployeeVehicle** | | |
| | Definition: | A vehicle operated by an employee of an organization that is used within that organization's ground |
| | Code: | 7 |
| **Value: FacilityVehicle** | | |
| | Definition: | A vehicle dedicated to a localized area within a private or restricted estate<br>Example: Facilities roads within an airport or theme park |
| | Code: | 8 |
| **Value: FarmVehicle** | | |
| | Definition: | vehicle commonly associated with farming activities |
| | Code: | 9 |
| **Value: HighOccupancyVehicle** | | |
| | Definition: | Vehicle populated with a number of occupants corresponding to (or exceeding) the specified minimum number of passengers |
| | Code: | 10 |
| **Value: LightRail** | | |
| | Definition: | train-like transport vehicle limited to a rail network within a limited area; does not include heavy rail lines |
| | Code: | 11 |
| **Value: MailVehicle** | | |
| | Definition: | A vehicle that collects, carries or delivers mail |
| | Code: | 12 |
| **Value: MilitaryVehicle** | | |
| | Definition: | Vehicle authorized by a military authority |
| | Code: | 13 |
| **Value: Moped** | | |
| | Definition: | Low powered two-wheeled motor vehicle with pedals |
| | Code: | 14 |
| **Value: Motorcycle** | | |
| | Definition: | High powered two-wheeled motor vehicle without pedal propulsion |
| | Code: | 15 |
| **Value: PassengerCar** | | |
| | Definition: | A small vehicle designed for private transport of people |
| | Code: | 16 |
| **Value: Pedestrian** | | |
| | Definition: | A person on foot |
| | Code: | 17 |
| **Value: PrivateBus** | | |
| | Definition: | A vehicle designed for transport of large groups of people, privately owned or chartered |
| | Code: | 18 |

| Class <<CodeList>>: VehicleType | | |
|---|---|---|
| **Value: PublicBus** | | |
| | Definition: | A vehicle designed for transport of large groups of people that is generally characterized by published routes and schedules |
| | Code: | 19 |
| **Value: ResidentalVehicle** | | |
| | Definition: | A vehicle whose owner is resident (or a guest) of particular street or town area |
| | Code: | 20 |
| **Value: SchoolBus** | | |
| | Definition: | Vehicle operated on behalf of a school to transport students |
| | Code: | 21 |
| **Value: SnowChainEquippedVehicle** | | |
| | Definition: | Any vehicle equipped with snow chains |
| | Code: | 22 |
| **Value: Tanker** | | |
| | Definition: | A truck with more than two axels used to transport liquid loads in bulk |
| | Code: | 23 |
| **Value: Taxi** | | |
| | Definition: | A vehicle licensed for hire usually fitted with a meter |
| | Code: | 24 |
| **Value: TransportTruck** | | |
| | Definition: | A truck vehicle for long range transport of goods |
| | Code: | 25 |
| **Value: TrolleyBus** | | |
| | Definition: | A bus-like mass transport vehicle hocked up to an electrical network for power supply |
| | Code: | 26 |
| **Value: VehicleForDisabledPerson** | | |
| | Definition: | A vehicle with supporting identification that designates a vehicle for disabled persons |
| | Code: | 27 |
| **Value: VehiclesWithSnowChainsOrSnowTyres** | | |
| | Definition: | A vehicle with snow chains or snow tyres |
| | Code: | 28 |

| Class <<DataType>>: FuzzyTimePeriod | | |
|---|---|---|
| | Definition: | A time period which is fuzzy |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: beginOrDuration** | | |
| | Definition: | Time period begins with a fuzzy time or the duration is a fuzzy time |
| | Value type: | FuzzyTime |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: endOrDuration** | | |
| | Definition: | time period ends with a fuzzy time or the duration is a fuzzy time |
| | Value type: | FuzzyTime |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |

| Class <<CodeList>>: FuzzyTime | | |
|---|---|---|
| | Definition: | Not precised time |
| | Status: | Proposed |

| Class <<CodeList>>: FuzzyTime | | |
|---|---|---|
| | Stereotypes: | |
| **Value: External** | | |
| | Definition: | Starting period controlled by external device [GDF 4] |
| | Code: | 1 |
| **Value: Dawn** | | |
| | Definition: | Starts at dawn [GDF 4] |
| | Code: | 2 |
| **Value: Dusk** | | |
| | Definition: | Starts at dusk [GDF 4] |
| | Code: | 3 |
| **Value: School** | | |
| | Definition: | Starts at any school period (date and hour) [GDF 4] |
| | Code: | 4 |
| **Value: Holiday** | | |
| | Definition: | Starts at any holiday [GDF 4] |
| | Code: | 5 |
| **Value: Winter** | | |
| | Definition: | Beginning of winter [GDF 4] |
| | Code: | 6 |
| **Value: Spring** | | |
| | Definition: | Beginning of spring [GDF 4] |
| | Code: | 7 |
| **Value: Summer** | | |
| | Definition: | Beginning of summer [GDF 4] |
| | Code: | 8 |
| **Value: Autumn** | | |
| | Definition: | Beginning of autumn [GDF 4] |
| | Code: | 9 |
| **Value: HighTide** | | |
| | Definition: | Beginning of high tide [GDF 4] |
| | Code: | 10 |
| **Value: LowTide** | | |
| | Definition: | Beginning of low tide [GDF 4] |
| | Code: | 11 |
| **Value: HighWater** | | |
| | Definition: | Beginning of high water [GDF 4] |
| | Code: | 12 |
| **Value: LowWater** | | |
| | Definition: | Beginning of low water [GDF 4] |
| | Code: | 13 |
| **Value: WetSeason** | | |
| | Definition: | Beginning of wet season [GDF 4] |
| | Code: | 14 |
| **Value: DrySeason** | | |
| | Definition: | Beginning of dry season [GDF 4] |
| | Code: | 15 |
| **Value: PeakHours** | | |
| | Definition: | Start of peak hours, peak hours include rush hour and activity/ scheduled event based times. These would vary by location and by season [GDF 4] |
| | Code: | 16 |
| **Value: OffPeakHours** | | |
| | Definition: | start of off-peak hours [GDF 4] |
| | Code: | 17 |

| Class <<CodeList>>: FuzzyTime | | |
|---|---|---|
| **Value: Day** | | |
| | Definition: | Start of day [GDF 4] |
| | Code: | 18 |
| **Value: Night** | | |
| | Definition: | Start of night [GDF 4] |
| | Code: | 19 |

| Class <<CodeList>>: TimePeriodType | | |
|---|---|---|
| | Definition: | Defines if the time period occurs once or are repeated |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: OccursOnce** | | |
| | Definition: | The time period occurs only one time |
| | Code: | 1 |
| **Value: Yearly** | | |
| | Definition: | The time period occurs one time a year |
| | Code: | 2 |
| **Value: Monthly** | | |
| | Definition: | The time period occurs one time a month |
| | Code: | 3 |
| **Value: Weekly** | | |
| | Definition: | The time period occurs one time a week |
| | Code: | 4 |
| **Value: Daily** | | |
| | Definition: | The time period occurs one time a day |
| | Code: | 5 |

| Class <<CodeList>>: WeekdayType | | |
|---|---|---|
| | Definition: | Defines where the week number is counted from |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: Absolute** | | |
| | Definition: | The weeks are counted from the beginning of a year |
| | Code: | 1 |
| **Value: FromEndOfMonth** | | |
| | Definition: | The weeks are counted from the end of a month |
| | Code: | 2 |
| **Value: FromStartOfMonth** | | |
| | Definition: | The weeks are counted from the beginning of a month |
| | Code: | 3 |

| Class <<CodeList>>: TimePeriodValidity | | |
|---|---|---|
| | Definition: | Defines if the time period is included or excluded |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: IncludePeriod** | | |
| | Definition: | The time period is included |
| | Code: | 1 |
| **Value: ExcludePeriod** | | |
| | Definition: | The time period is excluded |
| | Code: | 2 |

| Class <<DataType>>: IntegerInterval | | |
|---|---|---|
| | Definition: | Defines an integer interval with a start and a length |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: start** | | |
| | Definition: | The start of the interval (included) |
| | Value type: | Integer |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: length** | | |
| | Definition: | The length of the interval |
| | Value type: | Integer |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |

| Class <<DataType>>: TimeInterval | | |
|---|---|---|
| | Definition: | Defines a time interval |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: begin** | | |
| | Definition: | Starting time |
| | Value type: | Time [ISO 19103] |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: lengthSeconds** | | |
| | Definition: | Duration in seconds |
| | Value type: | Integer |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |

| Class <<CodeList>>: LoadType | | |
|---|---|---|
| | Definition: | Defines where the different types of loads |
| | Status: | Proposed |
| | Stereotypes: | <<CodeList>> |
| **Value: ExplosiveLoad** | | |
| | Definition: | Explosive load |
| | Code: | 1 |
| **Value: WaterPollutingLoad** | | |
| | Definition: | Water polluting load |
| | Code: | 2 |
| **Value: OtherDangerousLoad** | | |
| | Definition: | Other dangerous load |
| | Code: | 3 |

| Class <<CodeList>>: RoadCategoryType | | |
|---|---|---|
| | Definition: | Specifies the possible road category types |
| | Status: | Proposed |
| | Stereotypes: | CodeList |
| **Value: Motorway** | | |
| | Definition: | Motorway |
| | Code: | 1 |
| **Value: RoadWithDualCarriageway** | | |
| | Definition: | Road with dual carriageway |
| | Code: | 2 |

| Class <<CodeList>>: RoadCategoryType | | |
|---|---|---|
| **Value: RoadWithSingleCarriageway** | | |
| | Definition: | Road with single carriageway |
| | Code: | 3 |
| **Value: UrbanRoad** | | |
| | Definition: | Urban road/street |
| | Code: | 4 |
| **Value: ResidentialAreaRoad** | | |
| | Definition: | Residential area road/street |
| | Code: | 5 |
| **Value: OtherRoad** | | |
| | Definition: | Road of a category other than the above |
| | Code: | 6 |

## *6.4. Package Location referencing*

This package defines the classes needed for an AGORA-C compliant location referencing [3]. Observe that ROSATTE only specifies space for keeping a string that has been encoded using AGORA-C and base64 (to get a character encoding of an AGORA-C binary string).

ROSATTE uses AGORA-C with the following exception:

- Only functional road classed 0-4 shall be used and with the following interpretation:
  - FC 0: motorway, freeway (flow road)
  - FC 1: major road for motor vehicles not a motorway (through road)
  - FC 2: arterial road (both long distance and local)
  - FC 3: local distributor/collector road
  - FC 4: local road
- There is no 5 characters restriction in the road descriptor. Entire road names or numbers may be used

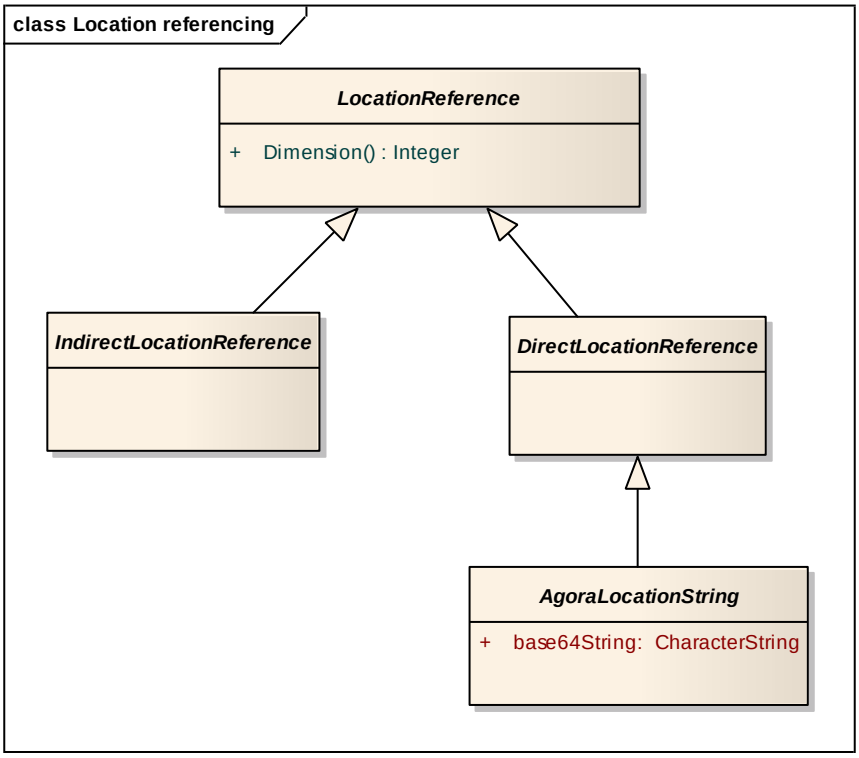



**Figure 8 - Classes for location referencing**

| **Class: LocationReference** | | |
|---|---|---|
| | Definition: | Abstract class that specifies any type of location reference for ROSATTE |
| | Subtype of: | |
| | Status: | Proposed |
| | Stereotypes: | |
| **Operation: Dimension** | | |
| | Definition: | Returns the dimension for the location reference.<br>0 : Point<br>1 : Linear<br>2 : Area (not used at the moment for ROSATTE)<br><br>Note:<br>The operation is currently only intended to be used to be able to specify the required location reference dimension for the various safety feature types in the application schema. |
| | Return type: | Integer |
| | Stereotypes: | |

| **Class: IndirectLocationReference** | | |
|---|---|---|
| | Definition: | Abstract class that at the moment serves as a placeholder for any type of indirect location reference that may be used in ROSATTE.<br><br>Note : Examples of indirect location references are:<br>- linear reference<br>- address<br>- location code<br><br>Note: Linear referencing is the preferred location referencing method for data associated with the road network in INSPIRE. This method may be de- |

| Class: IndirectLocationReference | | |
|---|---|---|
| | | fined as an option<br><br>Note: This kind of location reference is classified as “Pre-coded” in ISO 17572 |
| | Subtype of: | LocationReference |
| | Status: | Proposed |
| | Stereotypes: | |

| Class: DirectLocationReference | | |
|---|---|---|
| | Definition: | Abstract class that specifies all types of direct location references in accordance with ISO 19111.<br><br>Note: This kind of location reference is classified as “Dynamic” in ISO 17572 |
| | Subtype of: | LocationReference |
| | Status: | Proposed |
| | Stereotypes: | |

| Class: AgoraLocationString | | |
|---|---|---|
| | Definition: | A type of direct location reference that uses coordinates and core attribution according to the AGORA-C specification (ISO 17572-3) |
| | Subtype of: | DirectLocationReference |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: base64String** | | |
| | Definition: | The AGORA binary string encoded using base64 |
| | Value type: | CharacterString |
| | Multiplicity: | 1 |
| | Stereotypes: | |

## *6.5. Package Updates*

This package defines the various update primitives to be used when exchanging safety attribute updates.

The fundamental purpose of incremental updates is to save bandwidth, time and money with regards to data exchange. Instead of exchanging complete datasets, only the data needed to accomplish data synchronization should be exchanged. The classes in this package are designed to be used as safety feature attribution in such a way that each safety feature can signal the desired update event that occurred/shall occur.

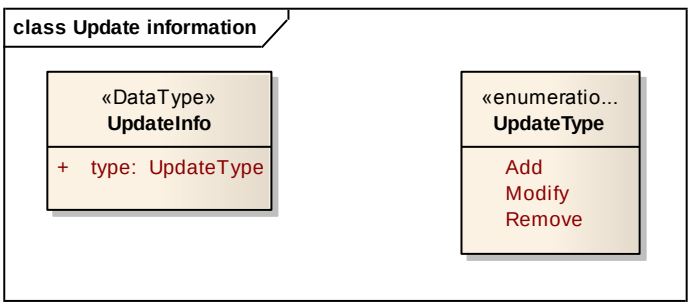



**Figure 9 - Update primitives**

### 6.5.1. Class definitions

| Class <<DataType>>: UpdateInfo | | |
|---|---|---|
| | Definition: | The information needed to represent a safety feature update. |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: type** | | |
| | Definition: | The type of update |
| | Value type: | UpdateType |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class <<enumeration>>: UpdateType | | |
|---|---|---|
| | Definition: | Defines the different update types |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: Add** | | |
| | Definition: | A new feature is inserted at the referenced location, without affecting any existing attribute feature |
| | Code: | 1 |
| **Value: Modify** | | |
| | Definition: | Signals a modification of the attributes of an existing safety feature referenced by an object identifier |
| | Code: | 2 |
| **Value: Remove** | | |
| | Definition: | Signals a removal of the existing safety feature referenced by an object identifier |
| | Code: | 3 |

## 6.6. Package Metadata

The metadata package describes the dataset metadata elements for ROSATTE. There are essentially two uses in ROSATTE for the metadata schema:

1. As a specification for the structure of a metadata store necessary in the support of discovery services
2. As a specification for the structure of metadata within an exchanged dataset. This metadata serves as a kind of attached delivery note identifying and describing the exchanged data and it's quality

We have already concluded that from the perspective of an enacting authority, there is no reason why ROSATTE should live under different rules and in a different environment than INSPIRE. The INSPIRE implementing rules will be mandatory for the enacting authorities anyway.

The listing below describes the ROSATTE use of metadata elements from the INSPIRE implementing rule for metadata [14]. The mapping of INSPIRE metadata elements to ISO 19115 [23] is described in the INSPIRE document "Draft Guidelines – INSPIRE metadata implementing rules based on ISO 19115 and ISO 19119" [15]. An XML encoding valid for the ROSATTE exchange format for metadata is found in ISO 19139 [28].

The INSPIRE Metadata element reference column contains the reference to the corresponding clause in the INSPIRE metadata implementing rule.

| INSPIRE<br>Metadata element reference | ROSATTE comment<br>(M) – Mandatory, (O) – Optional |
|---|---|
| 2.2.1.1 Resource title | Used for both discovery service (M) and dataset (M) |
| 2.2.1.2 Abstract | Used for both discovery service (M) and dataset (M) |
| 2.2.1.3 Resource type | Used for both discovery service (M) and dataset (M) |
| 2.2.1.4 Resource locator | Used for both discovery service (M) and dataset (M) |
| 2.2.1.5 Unique resource identifier | Only needed within the transmitted datasets (M). Needs to uniquely identify every exchanged dataset. Used by the feedback loop. |
| 2.2.1.6 Coupled resource | Only needed for discovery service (O) |
| 2.2.1.7 Resource language | **eng** (M) |
| 2.2.2.1 Topic category | **transportation** (M) |
| 2.2.3.1 Keyword value | **rosatte** (M)<br>This keyword shall be used for all data complying with this specification. |
| 2.2.3.2 Originating controlled vocabulary | Title: **ROSATTE – Specification of data exchange methods, D3.1, Version XX** (this document) (M)<br>Date.dateType : **publication** (M)<br>Date.date : **2009-mm-dd** (publication date of this document) (M) |
| 2.2.4.1 Geographic bounding box | Used for both discovery service (M) and dataset (M) |
| 2.2.5.1 Temporal extent | Used for both discovery service (M) and dataset (M) |
| 2.2.5.2 Date of publication | Used for dataset (M). Specifies when the dataset was published (may the same date as the creation date). |
| 2.2.5.3 Date of last revision | Used for both discovery service (M) and dataset (M). Specifies when the original data was revised. |
| 2.2.5.4 Date of creation | Used for both discovery service (M) and dataset (M). |
| 2.2.5.5 Alternate references | Not used in ROSATTE |
| 2.2.6.1 Lineage | Used for both discovery service (M) and dataset (O). |
| 2.2.6.2 Spatial resolution | Used for both discovery service (M) and dataset (O).<br>Example: 1 meter |
| 2.2.7.1 Conformity.Specification | Not used for ROSATTE. All datasets are assumed to conform to this specification |
| 2.2.7.2 Conformity.degree | Not used for ROSATTE. All datasets are assumed to conform to this specification |
| 2.2.8 Conditions applying to access and use | Used for both discovery service (O) and dataset (O). |
| 2.2.9 Limitations on public access | Used for both discovery service (O) and dataset (O). |
| 2.2.10.1 Responsible party | Used for both discovery service (M) and dataset (M). |
| 2.2.10.2 Responsible party role | Used for both discovery service (M) and dataset (M). |
| 2.2.11.1 Metadata point of contact | Only needed for discovery service (M) |
| 2.2.11.2 Metadata date | Only needed for discovery service (M) |
| 2.2.11.3 Metadata language | **eng** (M) |

**Table 11 - Metadata elements from INSPIRE**

The table below describes additional elements proposed for ROSATTE which are not part of the INSPIRE metadata implementing rule.

| ISO 19115 - metadata element | ROSATTE comment<br>(M) – Mandatory, (O) – Optional |
|---|---|
| Content information.feature catalogue description<br>❖ IncludedWithDataset<br>❖ featureTypes | Used for discovery service (M) to indicate the types of safety features that is supported by a certain enacting authority. |

| | |
|---|---|
| Identification<br>❖ characterSet | Used for discovery service (M). |
| Reference system | Used for discovery service (M) and also within the datasets (M) to specify the used reference system for the data. ROSATTE requires that **WGS84 (EPSG:4326)** is used. |
| Geographic location<br>- Vertical extent | Used for both discovery service (O) and dataset (O) |

**Table 12 - Additional ISO 19115 metadata elements for ROSATTE**

A metadata description of data quality is needed both on discovery services and dataset level. Data users need to find data that fits all their requirements including quality requirements (discovery services). When a customer receives data from a supplier, he needs to check whether this certain datum or dataset meets his requirements for data import (dataset level).
The quality metadata should be used to describe the quality of the received dataset. However, the quality of the individual datum (update message) cannot be measured at the time of the update generation. This quality evaluation is subject to a post processing assessment. Therefore the metadata can only contain a qualitative description of the data acquisition process.
These parameters have to be determined either in advance, e.g. by comparison with reference data or during the data acquisition process. It is planned to determine quality parameters during the project's test period.
The local authorities may have more than one data acquisition procedure because they maybe have different internal data sources or different responsibilities to acquire data. It is therefore not possible to describe the overall quality of the data source (i.e. local authority) as such. The idea is rather, that the data acquisition tools to be used in ROSATTE give the opportunity to define acquisition procedures and set their respective quality parameters. The operator who introduces a new or changed traffic regulation can then select the used method when he enters the new regulation to his database. The ROSATTE update message with its respective metadata description is then created automatically in the background.

Table 13 shows basic quality parameters to describe data quality. The parameters are mainly taken from ISO 19115 and their definitions can be found there. One additional element "Availability" has been added, however, since it is considered important. The definition for this element can be found in [1]:

Availability: *degree to which geographic data are available at a certain place and at a defined time*

It is foreseen to concretize these parameters during the preparation of the test period and to include the final parameters in the revised document after the test period.

| **quality metadata element** | **ROSATTE comment (M) - Mandatory, (O) - Optional** | **Corresponding ISO 19115 element** |
|---|---|---|
| Accuracy | M (Discovery/Dataset) | DQ_PositionalAccuracy |
| Consistency | M (Discovery/Dataset) | DQ_LogicalConsistency |
| Correctness | M (Discovery/Dataset) | DQ_ThematicAccuracy |
| Up-to-dateness | M (Discovery/Dataset) | DQ_TemporalAccuracy |
| Completeness | M (Discovery/Dataset) | DQ_Completeness |
| Availability | M (Discovery/Dataset) | Additional element |

**Table 13 - quality metadata elements for ROSATTE**

## *6.7. Package Feedback*

One fundamental purpose of the feedback loop is to provide a log file, from Map Providers back to the ROSATTE Data Service Operator, describing the outcome when handling the delivered road safety attributes.

A possible purpose of the feedback loop is to give the ROSATTE Data Service Operator a possibility to take appropriate actions according to the information retrieved from the received receipt. This might be to correct obvious technical mistakes or to send it further on to the Data Store operators and the Enacting authorities.

The incremental data that has been transferred from the ROSATTE Data Service Operator to the Map Provider will not be repeatedly transferred again (unless in a full supply).
This means that the Map Provider has to keep track of all received data regardless of if it was possible to handle it or not when first delivered.

This package defines the classes needed to support the feedback loop.

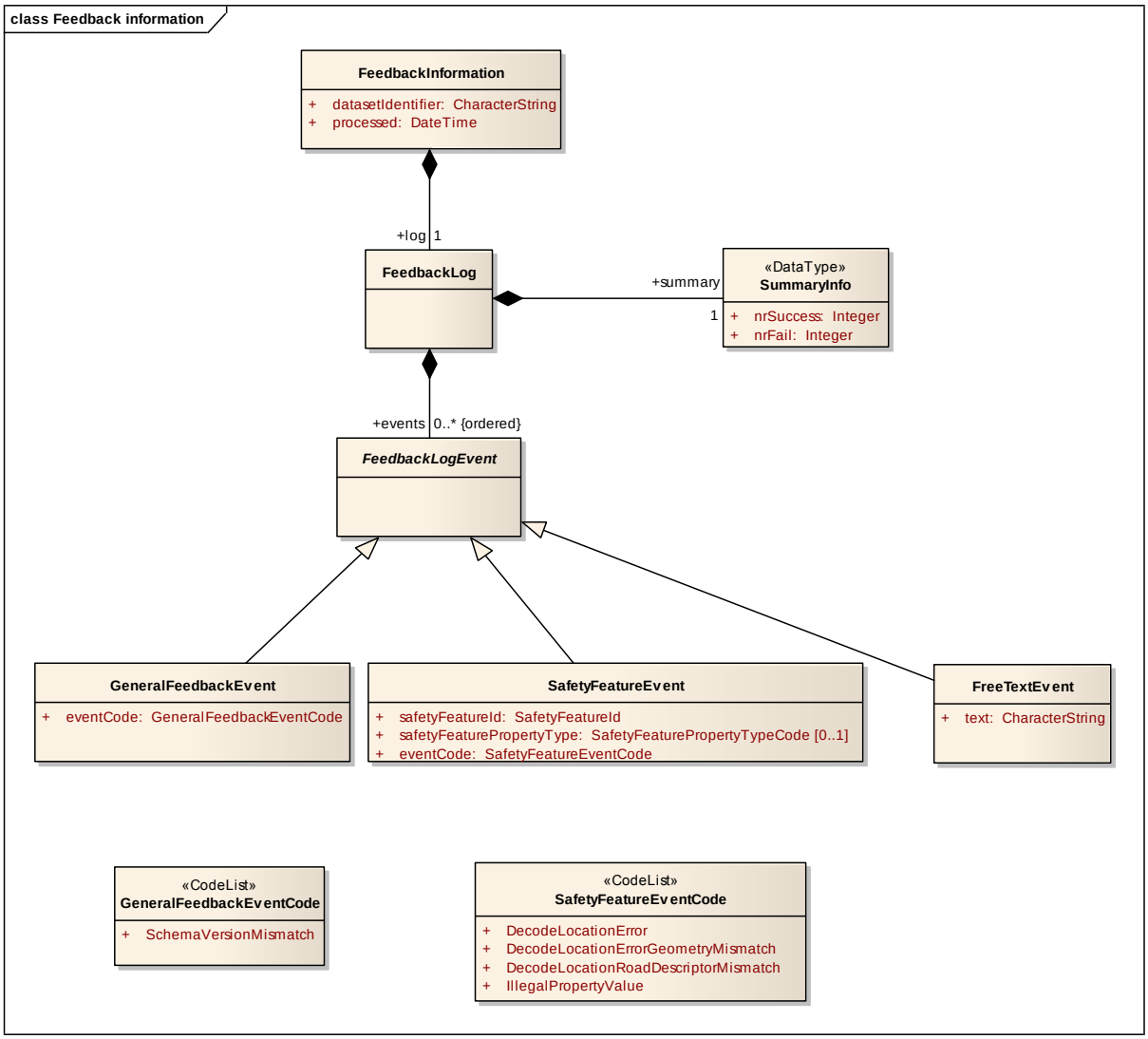



**Figure 10 - Feedback information**

| **Class: FeedbackInformation** | | |
|---|---|---|
| | Definition: | Represents the information sent back from a downloading client to the provider of safety attributes |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: datasetIdentifier** | | |
| | Definition: | The identification of the dataset which is the source of this feedback information. This identifier shall correspond to the metadata element 2.2.1.5 Unique resource identifier in the dataset for which the feedback is created. |
| | Value type: | CharacterString |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: processed** | | |
| | Definition: | Indicates when the downloaded dataset was processed at the client side |
| | Value type: | DateTime |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Association role: log | | |
|---|---|---|
| | Definition: | Refers to a log which records all the interesting events during download and integration of safety attributes. |
| | Value type: | FeedbackLog |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class: FeedbackLog | | |
|---|---|---|
| | Definition: | Represents a log which records all events that are reported back from a client as a result of download and integration of data |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | |
| **Association role: summary** | | |
| | Definition: | A result summary |
| | Value type: | SummaryInfo |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Association role: events** | | |
| | Definition: | The set of Refers to a log which records all the interesting events during download and integration of safety attributes. |
| | Value type: | FeedbackLogEvent |
| | Multiplicity: | 1..* |
| | Stereotypes: | |

| Class <<DataType>>: SummaryInfo | | |
|---|---|---|
| | Definition: | Information that summarizes a feedback result |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: nrSuccess** | | |
| | Definition: | Number of successfully decoded and integrated safety features. |
| | Value type: | Integer |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: nrFail** | | |
| | Definition: | Number of safety features for which integration failed. |
| | Value type: | Integer |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| Class: FeedbackLogEvent | | |
|---|---|---|
| | Definition: | Abstract base class that represents any type of event that may occur during download and integration of data that needs to be reported back to the provider |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | |

| Class: SafetyFeatureEvent | | |
|---|---|---|
| | Definition: | Represents one event that may occur during download and integration of data that needs to be reported back to the provider |
| | Subtype of: | FeedbackLogEvent |

| | | |
|---|---|---|
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: safetyFeatureId** | | |
| | Definition: | The identification of the safety feature that is concerned by the event. |
| | Value type: | SafetyFeatureId |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Attribute: safetyFeaturePropertyType** | | |
| | Definition: | The identification of the safety feature property type that is concerned by the event. If unset, this event does not concern any safety attribute property type in particular. |
| | Value type: | SafetyfeaturePropertyTypeCode |
| | Multiplicity: | 0..1 |
| | Stereotypes: | |
| **Attribute: eventCode** | | |
| | Definition: | Indicates the type of event that occurred |
| | Value type: | SafetyFeatureEventCode |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| **Class: generalFeedbackEvent** | | |
|---|---|---|
| | Definition: | Represents one event that may occur during download and integration of data that needs to be reported back to the provider |
| | Subtype of: | FeedbackLogEvent |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: eventCode** | | |
| | Definition: | Indicates the type of event that occurred |
| | Value type: | GeneralFeedbackEventCode |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| **Class: FreeTextEvent** | | |
|---|---|---|
| | Definition: | Represents one event that is described using free text |
| | Subtype of: | FeedbackLogEvent |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: text** | | |
| | Definition: | The text describing the event |
| | Value type: | CharacterString |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| **Class <<CodeList>>: SafetyFeatureEventCode** | | |
|---|---|---|
| | Definition: | Specifies an event code for safety feature related feedback events |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: DecodeLocationError** | | |
| | Definition: | An unspecified error when decoding the location of a safety feature |
| | Code: | 1 |
| **Value: DecodeLocationErrorGeometryMismatch** | | |
| | Definition: | A geometric mismatch error when decoding the location of a safety feature |
| | Code: | 2 |
| **Value: DecodeLocationErrorRoadDescriptorMismatch** | | |

| Class <<CodeList>>: SafetyFeatureEventCode | | |
|---|---|---|
| | Definition: | A road descriptor mismatch when decoding the location of a safety feature |
| | Code: | 3 |
| **Value: IllegalPropertyValue** | | |
| | Definition: | An illegal property value for the property type indicated by SafetyFeatureEvent.safetyFeaturePropertyType |
| | Code: | 4 |

| Class <<CodeList>>: SafetyFeatureEventCode | | |
|---|---|---|
| | Definition: | Specifies an generic event code for feedback events |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: SchemaVersionMismatch** | | |
| | Definition: | Occurs when the versions of the schemas of the server and client of safety feature data don't match |
| | Code: | 1 |

## *6.8. Package Dataset*

This package defines all the classes needed at dataset/message level such as:

- identification of a particular dataset/message
- other "top level" information to enforce a traceable and quality assured data chain

ROSATTE datasets may contain both datasets with incremental updates and datasets with snapshots of safety features. The difference between these types of datasets is:

- Datasets with incremental updates have the attribute "type" set to "Update". All safety features in the dataset have update information
- A snapshot dataset have the attribute "type" set to "Snapshot". No safety feature in the dataset have update information

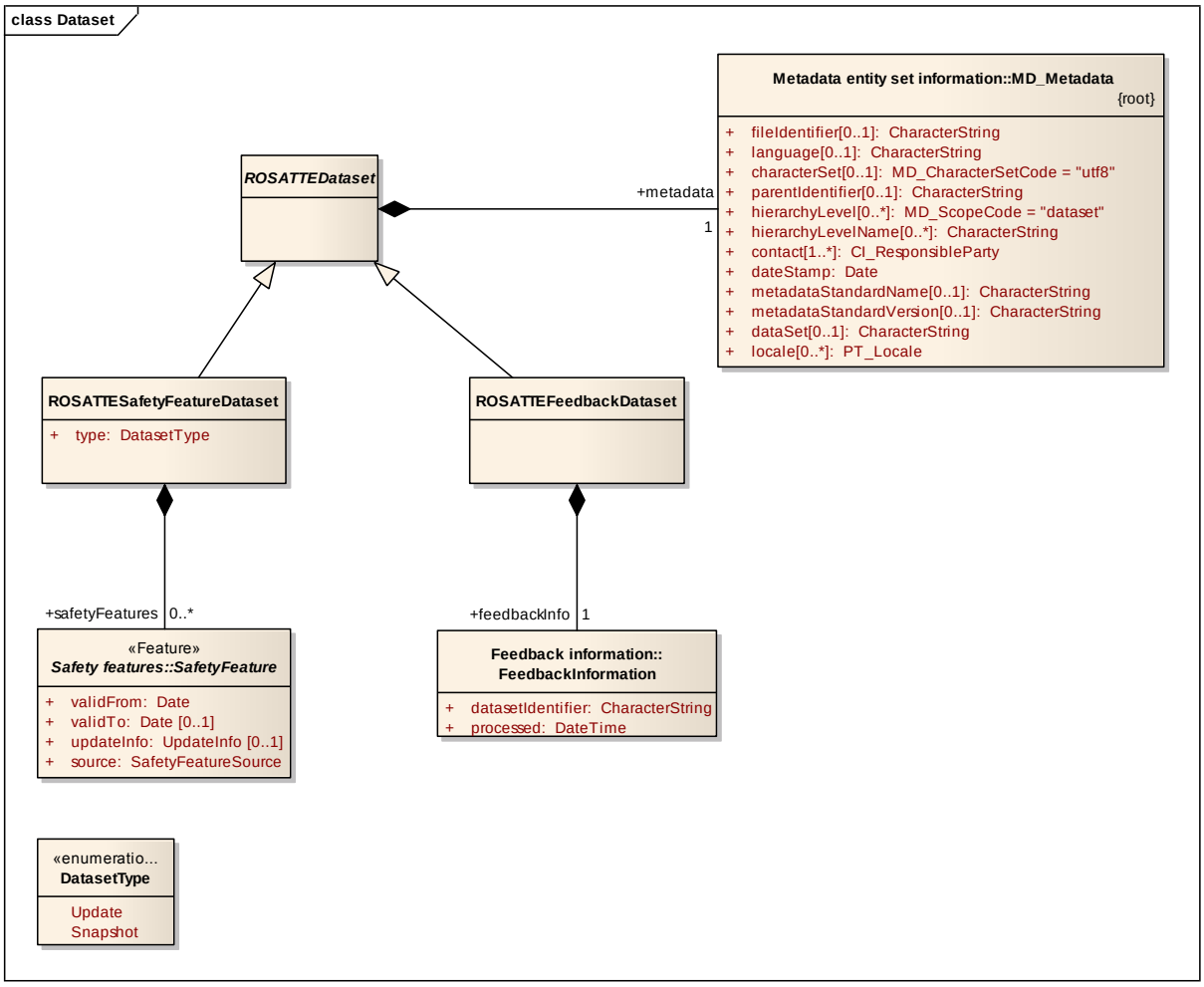



**Figure 11 - Dataset**

| **Class: ROSATTEDataset** | | |
|---|---|---|
| | Definition: | Abstract class that represents any type of dataset that may be exchanged using ROSATTE |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | |
| **Association role: metadata** | | |
| | Definition: | Metadata for the dataset.<br>Note: The metadata elements that may be used and rules for usage is specified in chapter 6.6 |
| | Value type: | MD_Metadata |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| **Class: ROSATTESafetyFeatureDataset** | | |
|---|---|---|
| | Definition: | A dataset containing safety feature with optional update information. |
| | Subtype of: | ROSATTEDataset |
| | Status: | Proposed |
| | Stereotypes: | |
| **Attribute: type** | | |
| | Definition: | Specifies the type of dataset |

| | | |
|---|---|---|
| | Value type: | DatasetType |
| | Multiplicity: | 1 |
| | Stereotypes: | |
| **Association role: safetyFeatures** | | |
| | Definition: | The set of safetyFeatures contained in this dataset |
| | Value type: | SafetyFeature |
| | Multiplicity: | 0..*<br>Note: One case where the dataset may be empty is when the dataset is a response to a request for updates when nothing actually did occur. |
| | Stereotypes: | |
| **Constraint: IncrementalUpdatesDataset**<br>Natural language:<br>All safety features in a dataset of type Update must contain update information<br>OCL:<br>type = **Update** implies<br>safetyFeatures->forall(s \| s.updateInfo->size() = 1) | | |
| **Constraint: SnapshotDataset**<br>Natural language:<br>All safety features in a dataset of type Snapshot must NOT contain update information<br>OCL:<br>type = **Snapshot** implies<br>safetyFeatures->forall(s \| s.updateInfo->size() = 0) | | |

| **Class: ROSATTEFeedbackDataset** | | |
|---|---|---|
| | Definition: | A dataset containing feedback to the enacting authority from an information-/map provider in response to data download and integration |
| | Subtype of: | ROSATTEDataset |
| | Status: | Proposed |
| | Stereotypes: | |
| **Association role: feedbackInfo** | | |
| | Definition: | The feedback information |
| | Value type: | FeedbackInformation |
| | Multiplicity: | 1 |
| | Stereotypes: | |

| **Class <<enumeration>>: DatasetType** | | |
|---|---|---|
| | Definition: | Specifies the different possible types of datasets for safety features |
| | Status: | Proposed |
| | Stereotypes: | |
| **Value: Update** | | |
| | Definition: | A dataset consisting of only incremental updates for safety features |
| | Code: | 1 |
| **Value: Snapshot** | | |
| | Definition: | A dataset consisting of a snapshot of safety features |
| | Code: | 2 |

# 7. Physical exchange format - structure and coding

## 7.1. Introduction

The primary role of the data exchange specification is to specify a coding for the various types of data listed in the previous chapter. The currently selected encoding for ROSATTE is ISO/OGC GML [27]. GML specifies rules for how to transform a UML application schema into a GML application schema which are used in this case. This coding shall be used whenever data flows between different actors within ROSATTE.

## 7.2. GML schema for ROSATTE

One xsd-file per UML package has been generated according to the following table:

| UML package | Xsd file |
|---|---|
| Dataset | Dataset.xsd |
| Update information | Update.xsd |
| Safety attributes | SafetyAttributes.xsd |
| Location referencing | LocationReferencing.xsd |
| Metadata | Metadata.xsd |
| Feedback information | Feedback.xsd |
| - | ROSATTE.xsd – main xsd that includes everything needed for a complete ROSATTE schema. |

**Table 14 - Mapping between UML packages and xsd files**

All xml elements defined by ROSATTE (and not reused from standardized schemas) will be defined in the **rst** namespace. The entire schema is listed in chapter 10.1. The major part of the xsd-files has been generated from uml by using the tool ShapeChange from Interactive Instruments [7] available under GNU GPL licensing.

# 8. Service specification

## 8.1. Introduction

So far we have specified a conceptual data model together with a concrete coding for that data model. In order to facilitate the actual data flow between the various actors within ROSATTE a number of services will have to be implemented. This chapter specifies these services.

The service specification of ROSATTE is inspired by INSPIRE Network services architecture.

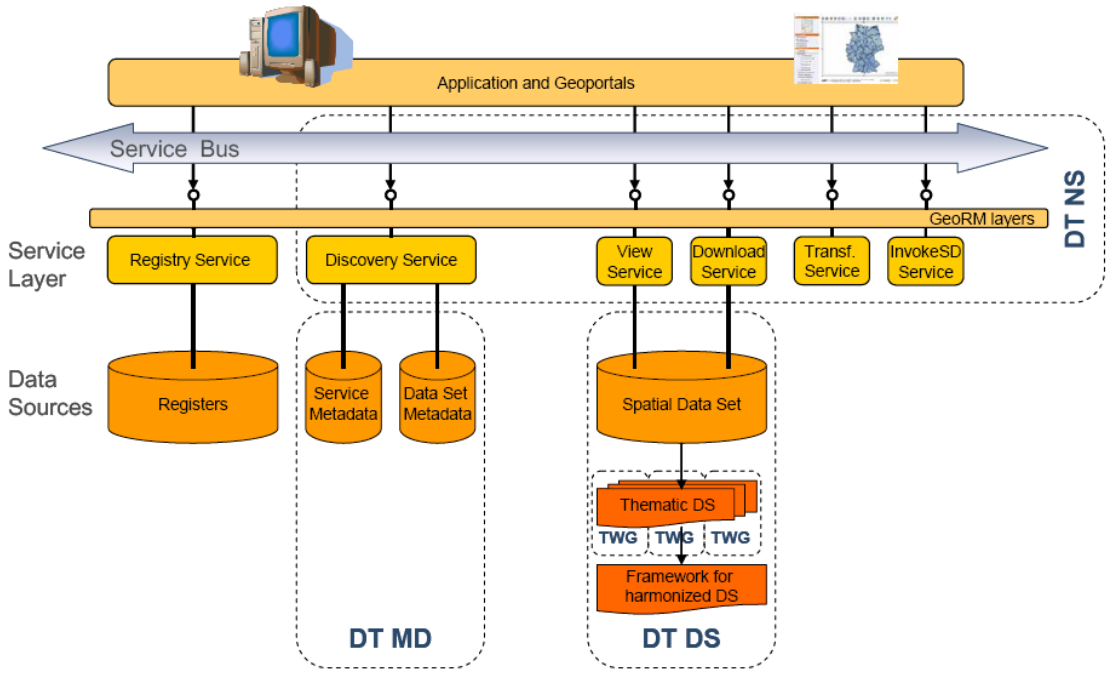



**Figure 12 - INSPIRE technical architecture overview**

The two parts of particular interest for ROSATTE are the download and discovery services, prioritized in that order.

### 8.1.1. Discovery service scope

The INSPIRE Directive asks Member States to "establish and operate a network of services" for the discovery of spatial data sets and services "for which metadata have been created". "Discovery services making it possible to search for spatial data sets and services on the basis of the content of the corresponding metadata and to display the content of the metadata." Within the geographic community various names have been assigned to instruments for discovering spatial data and services through the metadata properties; examples are Catalogue Services, Spatial Data Directory, Clearinghouse, Geographic Catalogue and Geodata Discovery Service. In INSPIRE these services are referred to as Discovery Services.

The goal of discovery is to support discovery, evaluation and use of spatial data and services through their metadata properties. Metadata is the information and documentation, which makes these resources understandable and sharable for users over time. Indexed and searchable metadata provide a controlled vocabulary against which discovery can be per-

formed. INSPIRE Discovery Services shall provide the functionality for users both to manage and search catalogues or the purpose of discovery and evaluation within the context of the INSPIRE Directive. The network of services should also include the technical possibility to enable public authorities to make their spatial datasets and services available. The INSPIRE Directive specifies that Member States shall ensure that public authorities are given the technical possibility to link their spatial datasets and services to the network. This 'linking' service is also offered in the context of a discovery service as a capability of the discovery service.

The definition of discovery services is not further pursued in this project due to the following reasons

- With the Inspire directive, member states are required to implement appropriate discovery services in all thematic areas of Inspire. Safety attributes, being either a part of road data or a separate theme, are clearly in the scope of such discovery services. It does therefore not make sense to define such services in parallel, rather future ROSATTE data providers shall ensure that their service information is accessible through the overarching discovery services which are being created as a result of the Inspire directive.
- As a subject of research such discovery services are not essential. Within the project, service providers and subscribers know each other and do not need to 'discover' each other.

### *8.1.2. Download service scope*

The INSPIRE Directive asks Member States to "establish and operate a network of download services, enabling copies of spatial data sets, or parts of such sets, to be downloaded and, where practicable, accessed directly". In addition, where public authorities levy charges for the download services, Member States shall ensure that e-commerce and GeoRM services are available.

A download service supports download of a complete dataset or datasets, or a part of a dataset or datasets, and where, practicable, provide direct access to complete datasets or parts of datasets. Gazetteer like services is also covered by a type of download service.
In the context of INSPIRE and the scope of the Implementing Rules, datasets are restricted to the categories defined by the Annexes I-III (see Article 4), and for which metadata exist and are updated according to Article 5, and that spatial datasets are interoperable and harmonized according to Article 7-10. It is worth to note that the conceptual or application schema of the **local or national** spatial data set may and will often differ from the **INSPIRE** harmonized specification of the spatial object types in the data specification. In this case a download service will transform queries and data between the application schema of the spatial dataset and the harmonized schema on-the-fly. Search criteria need to support searching based on spatial and temporal extents.

### *8.1.3. Download service decisions*

The way the download service makes update information available to clients can be varied in a lot of dimensions. One of these dimensions is "frequency". When regarding the frequency of the available updates the first choice to make is whether the update intervals can be freely chosen by the clients or whether the enacting authorities define these intervals. Another important definition to make is the versioning scheme to use. This has to be clearly defined to allow the client and the server agrees on the update data set to exchange.

The definitions made in this chapter refer to two aspects of the download service:

- client/server defined update intervals;
- the versioning scheme for the update data sets.

The following two sections present the basic differences between client- and server defined update intervals. After that a conclusion follows which defines which approach is chosen for the Rosatte infrastructure.

### *8.1.4. Approach 1: Server defined update intervals*

This approach lets the data store operator (public authority or road operator) define the update intervals. As a result the enacting authority can generate update data sets at predefined points in time. These update data sets can then be permanently stored on some media and made available to the clients without a need for the enacting authority to modify such a data set when requested.
The approach can be realized by defining fixed instances in time at which the current state of the database of an enacting authority is persisted. Such a persisted instance of the database is then compared to the previous persisted state. The difference between the two states is regarded as an update which can be made public via the download service.
A client which wants to update between more than two states has to apply each of the generated updates consecutively. An enhancement of this approach could be to combine a set of updates into one to create an update which corresponds to a longer time period.
The advantage of this approach is that the data handling on the server side is relatively simple. No changes of the database have to be stored separately.
The disadvantage of this approach is that the time periods for which updates are available cannot be chosen arbitrarily by the client but are defined by the servers' update policy. As a result these update intervals might not suit the needs of the users. However, as the data store operator is in general the authority which realizes and/or acquires the changes, it is anyway controlling the update process.

### *8.1.5. Approach 2: Client defined update intervals*

This approach lets the client define the update intervals. It allows a client to request an update for an arbitrary time period. Supporting such an update schema can only be done if the enacting authority is able to compute the content of such an update data set "on the fly" – that means in direct response to the request of the client.
The approach can be realized using a logging mechanism to store each modification of safety attribute data on the enacting authorities' database separately. Each of these modification operations can be qualified by a timestamp which stores the instance in time when the modification took place. The database might also store the identification of the user/public authority which modified the data. The modification information can be stored as a set of separate tables inside or beside of the spatial database. This data is denoted as "modification log". Apart from this modification log there also exists the normal state of the database which will also be changed by each modification operation. That means that the database always reflects the latest state of the safety attributes. The modification log can also be stored in a separate database or even in a plain file.
This approach allows generating updates for arbitrary time periods. As a result the client can decide which update granularity is best suited for its needs. Furthermore the performance of the spatial database will not be reduced by applying updates, as there is no accumulation scheme necessary to determine the most recent state. It is also possible to

store the current state of the database and the modification log on different systems. As a result this scheme tends to provide a better scalability of the application.

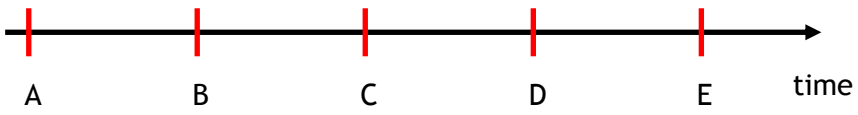
### *8.1.6. Server/Client defined update intervals - conclusion*

To make the requirements of the Rosatte infrastructure applicable to a wide set of database and GIS systems which are used at enacting authorities, the definition of update intervals is left to the server side – i.e. enacting authority. The disadvantage from the rigidity of predefined intervals for which update sets are available, can be compensated e.g. by sufficiently small time intervals. Also, for longer time periods into the past, the server may pre-compute and offer data sets which cover longer stretches *in addition* to offering data sets for each small time interval within this longer period.

### *8.1.7. Versioning scheme*

To allow the clients and servers which are part of the Rosatte infrastructure to agree about the content of a specific update data set a versioning scheme has to be defined. The versioning scheme has to be flexible enough to allow the enacting authorities to define update intervals which suit their needs and allows keeping up their specific workflows when managing safety attribute related data.
The versioning scheme of Rosatte defines how the "DatasetId" is generated which uniquely identifies each dataset which is made publicly available by an enacting authority. The DatasetId is represented by two 128 bit values where the first consists of an OSF/DCE UUID value identifying the enacting authority and the second consists of a high 64 bit part and a low 64 bit part. The high 64 bit part represents the start date of the update interval – the low part of represents the end date. Each of the 64 bit values contains the unix time of the specific start- or end date. The unix time is widely used an can therefore be regarded as a standard which can be easily adopted by Rosatte implementations. Note that the unix time is defined to be the number of seconds since 0:00:00 January 1$^{st}$, 1970 (UTC) – not including leap seconds. This time definition can also be found in the Java programming language and can easily be achieved using the C# programming language.

An enacting authority can decide on its own in which interval new updates are generated. Using the described versioning scheme it is not necessary that the enacting authority chooses equidistant intervals – it is also possible to define intervals in an ad hoc fashion. Another possibility is that update data sets are generated which aggregate a set of update intervals of smaller intervals.




The previous figure shows a possible regular update interval chosen by an enacting authority. In such a case the authority would make a set of update data sets available which could be described by the following four symbolical DatasetIds.

'AB', 'BC', 'CD', 'DE' …

Each character marks a unix time – i.e. the update data set defined by 'AB' starts at the unix time 'A' and ends at the unix time 'B'.

The enacting authority could also choose to additionally provide update data sets like 'AC', 'BD' or even 'AE'. The clients would be able to determine the contents of the update datasets by simply interpreting the DatasetId.

## *8.2. Interface specification*

This section identifies and specifies the interfaces needed for the communication between an information provider/map provider and the enacting authorities. The interface specification defines the abstract interfaces/protocols that may be translated to different implementation technologies such as SOAP or REST.

The two approaches previously discussed, respectively server and client intervals, relates to the frequency with which updates are made available by the server side. The fact that both approaches are to be supported has to be considered in the definition of the discovery service and the download service.

### *8.2.1. Server side components*

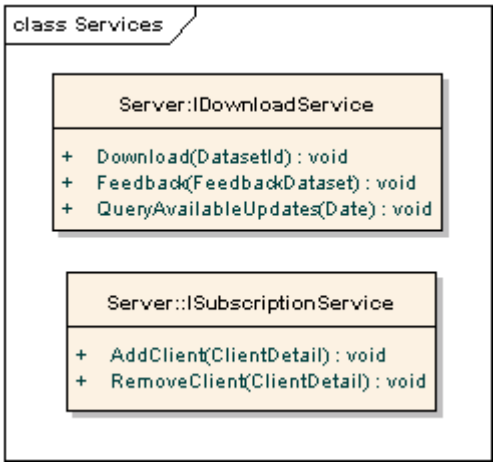



**Figure 13 - Server side interfaces**

| **Interface: IDownloadService** | | |
|---|---|---|
| | Definition: | Specifies the methods required for a ROSATTE download service |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | <<Interface>> |
| **Operation: QueryAvailableUpdates** | | |
| | Definition: | Called by the client to query which updates exist since a specific date in time. |
| | Returns: | A set of UpdateDescription structures. Each of this structures contains the following data:<br>- DatasetId (contains start date and end date as unix time) |
| | Parameters: | In Date: Specifies the start time of the requested update information. The client can define a lower bound for the time to which the update information applies. The data is specified as unix time in 64 bit unsigned encoding. |
| **Operation: Download** | | |

| | | |
|---|---|---|
| | Definition: | Called by the client to request and download data |
| | Returns: | Dataset : The dataset identification and the actual data as a GML stream according to the ROSATTE GML schema |
| | Parameters: | In DatasetId : uniquely defines the update dataset as defined in the return value of “QueryAvailableUpdates”. |
| **Operation: Feedback** | | |
| | Definition: | Called by the client to provide feedback connected to a particular download. |
| | Returns: | - |
| | Parameters: | In FeedbackDataset : The dataset which contain the feedback information. |

### *8.2.2. Client side components*

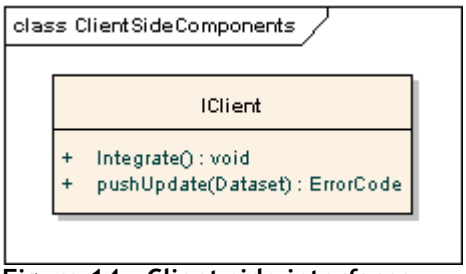



**Figure 14 - Client side interfaces**

| **Interface: IClient** | | |
|---|---|---|
| | Definition: | Specifies the methods required for a ROSATTE subscription client |
| | Subtype of: | - |
| | Status: | Proposed |
| | Stereotypes: | <<Interface>> |
| **Operation: Integrate** | | |
| | Definition: | Called by the server to notify clients that updated data exists |
| | Returns: | |
| | Parameters: | In EventType : specifies what event actually occurred on the server |
| **Operation: pushUpdate** | | |
| | Definition: | Called by the server to push an update dataset to the client. |
| | Returns: | ErrorCode : Describes the error which occurred during this call – if any. |
| | Parameters: | In DataSet : the update dataset which contains the updated safety attribute data. |

## *8.3. Use scenarios*

This section illustrates the various scenarios which are possible to use the defined services.

The first scenario describes the situation when a client makes a subscription for updates using the subscription service. This subscription tells the server that a specific client is interested in receiving update data. When the server has generated a new update dataset it calls the “pushUpdate” function of the client. The update dataset is transferred to the client as part of this call. The dataset is then integrated into the client’s database.

### 8.3.1. Scenario : Subscription and download (Push)

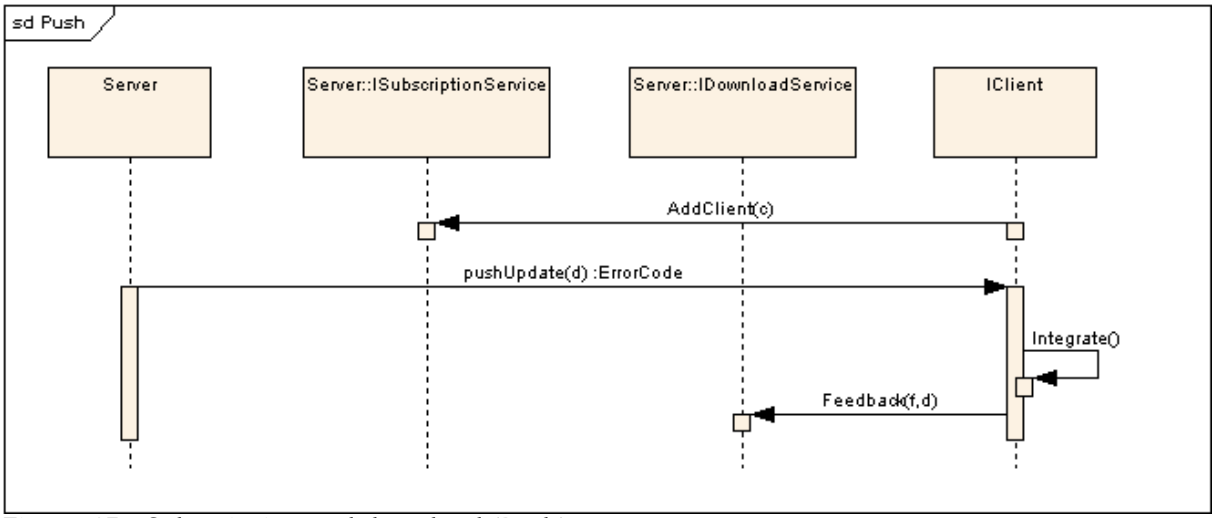



**Figure 15 – Subscription and download (Push)**

After integration in the client database the client shall provide feedback on the outcome of the process using the “Feedback” function of the download service.

The second scenario describes the situation when the client queries the server for update information using the function “QueryAvailableUpates” of the download service. The result of this call tells the client whether there is update information available for its own database. When the client has determined that there is update information, it uses the “Download” function of the download service. The returned dataset is then integrated into its database. As in the first scenario feedback information about the integration process is then transferred to the download service using the “Feedback” function.

### 8.3.2. Scenario : Query and download (Pull)

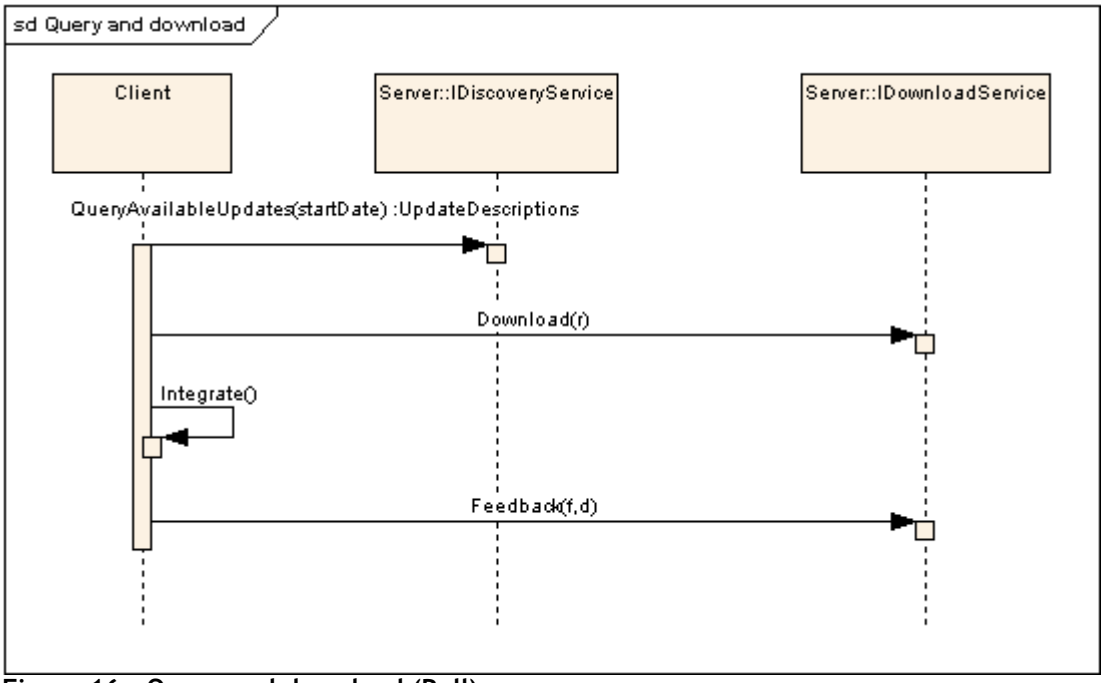



**Figure 16 - Query and download (Pull)**

## 8.4. Interface implementation specification

This section will contain the actual REST interface specification. It will be filled once the data exchange mechanisms have been tested and verified.

# 9. References


[1] AGORA-C, 2006a. Hendriks, T., Wevers, K., Hessling, M., and H.-W. Pfeiffer, "Specification of the AGORA-C on-the-fly location referencing method", version 1.0, 06 April 2005.

[2] AGORA-C, 2006b. Wevers, K., and Hendriks, T., "AGORA-C map-based location referencing", Transportation Research Record, 2006, vol. 1972, pp 115-122.

[3] AGORA-C, 2007. ISO, "Intelligent Transport System (ITS) - Location Referencing for Geographic Data-bases - Part 3: Dynamic Location References (Dynamic Profile)", ISO 17572-3, 08-12-2008, International Organization for Standardization (ISO), Geneva.

[4] GDF 4.0. International Standards Organisation (ISO), "Intelligent Transport Systems - Geographic Data Files (GDF) - Overall Data Specification", International Standard ISO 14825:2004, 03-02-2004 (GDF 4.0).

[5] MAPS&ADAS, 2004. MAPS&ADAS Consortium, S. T'Siobbel (ed.), et al., "Safety Digital Maps Requirements", Deliverable 12.31, version 10, 17 September 2004.

[6] MAPS&ADAS, 2007. MAPS&ADAS Consortium, K. Wevers (ed.), et al., "Specification and exchange format for safety related map information", Deliverable 12.41, version 09, 24 May 2007.

[7] http://www.interactive-instruments.de/index.php?id=28&L=1

[8] Oxford. Oxford Advanced Learner's Dictionary, on-line version.

[9] ROSATTE, 2007. ROSATTE Consortium, "ROad Safety ATTributes exchange infrastructure in Europe, ROSATTE, Annex 1 - Description of Work, annex to the contract of the ROSATTE Consortium with the European Commission grant agreement 213467), Brussels, 10 December 2007.

[10] TMC Forum, 2004. TMC Forum. Precise Location Referencing. TMC Forum Specification, version 05, Brussels, 20 September 2004.

[11] TMC Part 3. CEN ISO, "Traffic and Travel Information (TTI) - TTI Messages via traffic message coding - Part 3: Location referencing for ALERT- C", EN ISO 14819-3:2003, European Committee for Standardization (CEN), Brussels, and International Organization for Standardization (ISO), Geneva.

[12] UN, 1993. United Nations Economic Commission for Europe, Inland Transport Committee, "Convention on road traffic", done at Vienna on 8 November 1968, Amendment 1 (Incorporating the amendments to the Convention which entered into force on 3 September 1993), 1993.

[13] UN, 1995. United Nations Economic Commission for Europe, Inland Transport Committee, "Convention on road signs and signals", done at Vienna on 8 November 1968, Amendment 1 (Incorporating the amendments to the Convention which entered into force on 30 November 1995), 1995.

[14] INSPIRE Draft Implementing Rules for Metadata (Version 3), http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/INSPIRE_Metadata_ImplementingRule_v3_20071026.pdf

[15] Draft Guidelines – INSPIRE metadata implementing rules based on ISO 19115 and ISO 19119, http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/metadata/Draft_Guidelines_INSPIRE_metadata_implementing_rules.pdf

[16] D2.8.I.7 INSPIRE Data Specification on Transport Networks – Draft Guidelines http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/DataSpecifications/INSPIRE_DataSpecification_TN_v2.0.pdf

[17] ROSATTE, deliverable D1.1 State of the art

[18] ROSATTE, deliverable D1.2 & D1.3

[19] INSPIRE Generic Conceptual Model, D2.5
http://www.ec-gis.org/inspire/reports/ImplementingRules/DataSpecifications/D2.5_v3.0.pdf

[20] INSPIRE Guidelines for the encoding of spatial data, D2.7
http://www.ec-gis.org/inspire/reports/ImplementingRules/DataSpecifications/inspire_dataspec_D2.7_v2.0.pdf

[21] INSPIRE Draft Implementing Rules for Discovery and View Services, D3.7
http://www.ec-gis.org/inspire/reports/ImplementingRules/network/D3.7_Draft_IR_Discovery_and_View_Services_v2.0.pdf

[22] INSPIRE Network Services Architecture, D3.5
http://www.ec-gis.org/inspire/reports/ImplementingRules/network/D3_5_INSPIRE_NS_Architecture_v3-0.pdf

[23] ISO 19115:2003 Geographic information – Metadata

[24] ISO 19103 Geographic information – Conceptual schema language

[25] ISO 19108 Geographic information - Temporal schema

[26] ISO 19109 Geographic information – Rules for application schema

[27] ISO 19136 Geographic information – Geography markup language

[28] ISO 19139 geographic information – Metadata – XML schema implementation

[29] SOAP, Simple Object Access Protocol, http://www.w3.org/TR/soap/

[30] WSDL, Web Services Description Language, http://www.w3.org/TR/wsdl

[31] OpenGIS Catalogue Service Implementation Specification,
http://www.opengeospatial.org/standards/cat

[32] EuroRoadS Deliverable D6.8, Metadata Catalogue,
http://www.euroroads.org/php/Reports/D6.8 Metadata catalogue.pdf

[33] UML, Unified modeling language,
http://www.omg.org/technology/documents/formal/uml.htm

[34] REST, Representational state transfer,
http://en.wikipedia.org/wiki/Representational_State_Transfer

[35] EuroRoadS Deliverable D6.5, Core European Road Data,
http://www.euroroads.org/php/Reports/D6.5 Final specifikation of core European road data.pdf

# 10. Appendices

## *10.1. ROSATTE xsd schemas*

This section contains the formal xsd definitions for the ROSATTE physical exchange format. It is needed for the practical testing within the ROSATTE project and for future use of ROSATTE. The creation of xsd schemas is dependent on the UML schemas in chapter 6 and the encoding rules of GML [27]. References to schema files will change in the final version of this document when they exist in a stable location on the internet.

### *10.1.1. ROSATTE.xsd*

```
<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
    <annotation>
      <documentation>Main schema file for the ROSATTE exchange schema, version
1.0</documentation>
    </annotation>
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <include schemaLocation="LocationReferencing.xsd"/>
    <include schemaLocation="Update.xsd"/>
    <include schemaLocation="Conditions.xsd"/>
    <include schemaLocation="SafetyFeatures.xsd"/>
    <include schemaLocation="Feedback.xsd"/>
    <include schemaLocation="Dataset.xsd"/>
</schema>
```

### *10.1.2. Dataset.xsd*

```
<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:gmd="http://www.isotc211.org/2005/gmd" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
    <include schemaLocation="SafetyFeatures.xsd"/>
    <include schemaLocation="Feedback.xsd"/>
    <import namespace="http://www.w3.org/1999/xlink" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/xlink/xlinks.xsd"/>
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <import namespace="http://www.isotc211.org/2005/gmd" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gmd/gmd.xsd"/>
    <!--XML Schema document created by ShapeChange-->
    <element name="ROSATTEDataset" type="rst:ROSATTEDatasetType" abstract="true" substitution-
Group="gml:AbstractFeatureCollection"/>
    <complexType name="ROSATTEDatasetType" abstract="true">
      <complexContent>
        <extension base="gml:AbstractFeatureCollectionType">
          <sequence>
            <element name="metadata" type="gmd:MD_Metadata_Type" minOccurs="0"/>
          </sequence>
```

```
            </extension>
        </complexContent>
    </complexType>
    <complexType name="ROSATTEDatasetPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:ROSATTEDataset"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="ROSATTESafetyFeatureDataset" type="rst:ROSATTESafetyFeatureDatasetType"
substitutionGroup="rst:ROSATTEDataset"/>
    <complexType name="ROSATTESafetyFeatureDatasetType">
        <complexContent>
            <extension base="rst:ROSATTEDatasetType">
                <sequence>
                    <!--element name="safetyFeatures" type="rst:SafetyFeaturePropertyType" minOc-
curs="0" maxOccurs="unbounded"/-->
                    <element name="type" type="rst:DatasetTypeType"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="ROSATTESafetyFeatureDatasetPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:ROSATTESafetyFeatureDataset"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <simpleType name="DatasetTypeType">
        <restriction base="string">
            <enumeration value="Update"/>
            <enumeration value="Snapshot"/>
        </restriction>
    </simpleType>
    <element name="ROSATTEFeedbackDataset" type="rst:ROSATTEFeedbackDatasetType" substitu-
tionGroup="rst:ROSATTEDataset"/>
    <complexType name="ROSATTEFeedbackDatasetType">
        <complexContent>
            <extension base="rst:ROSATTEDatasetType">
                <sequence>
                    <element name="feedbackInfo" type="rst:FeedbackInformationPropertyType"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="ROSATTEFeedbackDatasetPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:ROSATTEFeedbackDataset"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
</schema>
```

### *10.1.3. SafetyFeatures.xsd*

```
<?xml version="1.0" encoding="windows-1252"?>
```

```
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
    <include schemaLocation="Update.xsd"/>
    <include schemaLocation="LocationReferencing.xsd"/>
    <include schemaLocation="Conditions.xsd"/>
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <!--XML Schema document created by ShapeChange-->
    <simpleType name="SafetyFeatureTypeCodeType">
      <union memberTypes="rst:SafetyFeatureTypeCodeEnumerationType
rst:SafetyFeatureTypeCodeOtherType"/>
    </simpleType>
    <simpleType name="SafetyFeatureTypeCodeEnumerationType">
      <restriction base="string">
          <enumeration value="PedestrianCrossing"/>
          <enumeration value="RestrictionForVehicles"/>
          <enumeration value="SpeedLimit"/>
          <enumeration value="StartOfSpeedLimit"/>
          <enumeration value="EndOfSpeedLimit"/>
          <enumeration value="ProhibitionOfOvertaking"/>
          <enumeration value="StartOfProhibitionOfOvertaking"/>
          <enumeration value="UseOfAudibleWarningDevicesProhibited"/>
          <enumeration value="StartOfUseOfAudibleWarningDevicesProhibited"/>
          <enumeration value="EndOfProhibitionOrRestriction"/>
          <enumeration value="ProhibitedTurn"/>
          <enumeration value="PassingWithoutStoppingProhibited"/>
          <enumeration value="Motorway"/>
          <enumeration value="StartOfMotorway"/>
          <enumeration value="EndOfMotorway"/>
          <enumeration value="NoEntry"/>
          <enumeration value="ClosedToAllVehiclesInBothDirections"/>
          <enumeration value="DirectionToBeFollowed"/>
          <enumeration value="SnowChainsCompulsory"/>
          <enumeration value="CompulsoryDirectionForVehicleCarryingDangerousGoods"/>
          <enumeration value="RoadForMotorVehicles"/>
          <enumeration value="StartOfRoadForMotorVehicles"/>
          <enumeration value="EndOfRoadForMotorVehicles"/>
          <enumeration value="BuiltUpArea"/>
          <enumeration value="StartOfBuiltUpArea"/>
          <enumeration value="EndOfBuiltUpArea"/>
          <enumeration value="ResidentialArea"/>
          <enumeration value="StartOfResidentialArea"/>
          <enumeration value="EndOfResidentialArea"/>
          <enumeration value="WarningSign"/>
      </restriction>
    </simpleType>
    <simpleType name="SafetyFeatureTypeCodeOtherType">
      <restriction base="string">
          <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
    <simpleType name="SafetyFeaturePropertyTypeCodeType">
      <union memberTypes="rst:SafetyFeaturePropertyTypeCodeEnumerationType
rst:SafetyFeaturePropertyTypeCodeOtherType"/>
```

```
    </simpleType>
    <simpleType name="SafetyFeaturePropertyTypeCodeEnumerationType">
      <restriction base="string">
          <enumeration value="MaximumSpeedLimit"/>
          <enumeration value="MinimumSpeedLimit"/>
          <enumeration value="RecommendedMaximumSpeedLimit"/>
          <enumeration value="RecommendedMinimumSpeedLimit"/>
          <enumeration value="MaximumWeightPerSingleAxle"/>
          <enumeration value="MaximumHeight"/>
          <enumeration value="MaximumWidth"/>
          <enumeration value="MaximumLadenWeight"/>
          <enumeration value="MaximumLength"/>
          <enumeration value="MaximumDistanceBetweenVehicles"/>
          <enumeration value="WarningSignType"/>
          <enumeration value="Controlled"/>
      </restriction>
    </simpleType>
    <simpleType name="SafetyFeaturePropertyTypeCodeOtherType">
      <restriction base="string">
          <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
    <simpleType name="WarningSignTypeType">
      <union memberTypes="rst:WarningSignTypeEnumerationType
rst:WarningSignTypeOtherType"/>
    </simpleType>
    <simpleType name="WarningSignTypeEnumerationType">
      <restriction base="string">
          <enumeration value="Danger"/>
          <enumeration value="DangerousIntersection"/>
          <enumeration value="DangerousCurve"/>
          <enumeration value="Slope"/>
          <enumeration value="UnevenRoadSurface"/>
          <enumeration value="SnowIceOrBlackIce"/>
          <enumeration value="SlipperyRoad"/>
          <enumeration value="Rockfall"/>
          <enumeration value="GravelOnTheRoad"/>
          <enumeration value="SideWind"/>
          <enumeration value="NarrowingRoad"/>
          <enumeration value="ConstructionWork"/>
          <enumeration value="Congestion"/>
          <enumeration value="TwoWayTraffic"/>
          <enumeration value="MovingBridge"/>
          <enumeration value="TrafficLightsAhead"/>
          <enumeration value="ChildrenPlaying"/>
          <enumeration value="PedestriansCrossing"/>
          <enumeration value="CyclistsCrossing"/>
          <enumeration value="AnimalsCrossing"/>
          <enumeration value="AirTraffic"/>
          <enumeration value="BusStops"/>
          <enumeration value="RailwayCrossing"/>
          <enumeration value="MooseCrossing"/>
      </restriction>
    </simpleType>
    <simpleType name="WarningSignTypeOtherType">
      <restriction base="string">
          <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
```

```
    <element name="SafetyAttributePropertyType" type="rst:SafetyAttributePropertyTypeType"
substitutionGroup="gml:AbstractGML"/>
    <complexType name="SafetyAttributePropertyTypeType">
      <complexContent>
        <extension base="gml:AbstractGMLType">
          <sequence>
            <element name="EAID_1AA56D56_F5FD_4f5a_919E_CF38BC520604.SE"
type="rst:SafetyFeaturePropertyValuePropertyType" minOccurs="0" maxOccurs="unbounded"/>
            <element name="unitOfmeasure" type="gml:UnitOfMeasureType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="SafetyAttributePropertyTypePropertyType">
      <sequence minOccurs="0">
        <element ref="rst:SafetyAttributePropertyType"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="SafetyFeature" type="rst:SafetyFeatureType" abstract="true" substitution-
Group="gml:AbstractFeature"/>
    <complexType name="SafetyFeatureType" abstract="true">
      <complexContent>
        <extension base="gml:AbstractFeatureType">
          <sequence>
            <element name="id" type="rst:SafetyFeatureIdPropertyType"/>
            <element name="locationReference" type="rst:LocationReferencePropertyType" mi-
nOccurs="0"/>
            <element name="validFrom" type="date"/>
            <element name="validTo" type="date" minOccurs="0"/>
            <element name="updateInfo" type="rst:UpdateInfoPropertyType" minOccurs="0"/>
            <element name="source" type="rst:SafetyFeatureSourceType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="SafetyFeaturePropertyType">
      <sequence minOccurs="0">
        <element ref="rst:SafetyFeature"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <simpleType name="SafetyFeatureSourceType">
      <annotation>
        <documentation>Value list for speedLimitSource.
[Euroroads]</documentation>
      </annotation>
      <restriction base="string">
        <enumeration value="FixedTrafficSign"/>
        <enumeration value="VariableTrafficSign"/>
        <enumeration value="Regulation"/>
        <enumeration value="TemporarySafetyFeature"/>
        <enumeration value="Other"/>
      </restriction>
```

```
    </simpleType>
    <element name="GenericSafetyFeature" type="rst:GenericSafetyFeatureType" substitution-
Group="rst:SafetyFeature"/>
    <complexType name="GenericSafetyFeatureType">
      <complexContent>
        <extension base="rst:SafetyFeatureType">
          <sequence>
            <element name="type" type="rst:SafetyFeatureTypeCodeType"/>
            <element name="properties" type="rst:SafetyFeaturePropertyValuePropertyType"
minOccurs="0" maxOccurs="unbounded"/>
            <element name="condition" type="rst:ConditionPropertyType" minOccurs="0"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="GenericSafetyFeaturePropertyType">
      <sequence minOccurs="0">
        <element ref="rst:GenericSafetyFeature"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="SafetyFeatureId" type="rst:SafetyFeatureIdType" substitution-
Group="gml:AbstractObject"/>
    <complexType name="SafetyFeatureIdType">
      <sequence>
        <element name="providerId" type="string"/>
        <element name="id" type="string"/>
      </sequence>
    </complexType>
    <complexType name="SafetyFeatureIdPropertyType">
      <sequence>
        <element ref="rst:SafetyFeatureId"/>
      </sequence>
    </complexType>
    <element name="SafetyFeaturePropertyValue" type="rst:SafetyFeaturePropertyValueType" subs-
titutionGroup="gml:AbstractObject"/>
    <complexType name="SafetyFeaturePropertyValueType">
      <sequence>
        <element name="type" type="rst:SafetyFeaturePropertyTypeCodeType"/>
        <element name="propertyValue" maxOccurs="unbounded"/>
      </sequence>
    </complexType>
    <complexType name="SafetyFeaturePropertyValuePropertyType">
      <sequence>
        <element ref="rst:SafetyFeaturePropertyValue"/>
      </sequence>
    </complexType>
</schema>
```

### *10.1.4. Conditions.xsd*

```
<?xml version="1.0" encoding="windows-1252"?>
<!-- edited with XMLSpy v2005 rel. 3 U (http://www.altova.com) by Lars Bergström (Triona AB) -->
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
```

```
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <!--XML Schema document created by ShapeChange-->
    <element name="LoadCondition" type="rst:LoadConditionType" substitution-
Group="rst:Condition"/>
    <complexType name="LoadConditionType">
      <complexContent>
        <extension base="rst:ConditionType">
          <sequence>
            <element name="loadType" type="rst:LoadTypeType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="LoadConditionPropertyType">
      <sequence>
        <element ref="rst:LoadCondition"/>
      </sequence>
    </complexType>
    <element name="AreaCondition" type="rst:AreaConditionType" substitution-
Group="rst:Condition"/>
    <complexType name="AreaConditionType">
      <complexContent>
        <extension base="rst:ConditionType">
          <sequence>
            <element name="areaType" type="rst:AreaTypeType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="AreaConditionPropertyType">
      <sequence>
        <element ref="rst:AreaCondition"/>
      </sequence>
    </complexType>
    <simpleType name="ConditionOperatorType">
      <restriction base="string">
        <enumeration value="OR"/>
        <enumeration value="XOR"/>
        <enumeration value="AND"/>
      </restriction>
    </simpleType>
    <element name="Condition" type="rst:ConditionType" abstract="true" substitution-
Group="gml:AbstractObject"/>
    <complexType name="ConditionType" abstract="true">
      <sequence>
        <element name="negate" type="boolean" minOccurs="0"/>
      </sequence>
    </complexType>
    <complexType name="ConditionPropertyType">
      <sequence>
        <element ref="rst:Condition"/>
      </sequence>
    </complexType>
    <element name="RoadCategoryCondition" type="rst:RoadCategoryConditionType" substitution-
Group="rst:Condition"/>
```

```
    <complexType name="RoadCategoryConditionType">
      <complexContent>
        <extension base="rst:ConditionType">
          <sequence>
            <element name="roadCategory" type="rst:RoadCategoryTypeType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="RoadCategoryConditionPropertyType">
      <sequence>
        <element ref="rst:RoadCategoryCondition"/>
      </sequence>
    </complexType>
    <element name="IntegerInterval" type="rst:IntegerIntervalType" substitution-
Group="gml:AbstractObject"/>
    <complexType name="IntegerIntervalType">
      <sequence>
        <element name="start" type="integer"/>
        <element name="length" type="integer" minOccurs="0"/>
      </sequence>
    </complexType>
    <complexType name="IntegerIntervalPropertyType">
      <sequence>
        <element ref="rst:IntegerInterval"/>
      </sequence>
    </complexType>
    <simpleType name="RoadCategoryTypeType">
      <union memberTypes="rst:RoadCategoryTypeEnumerationType
rst:RoadCategoryTypeOtherType"/>
    </simpleType>
    <simpleType name="RoadCategoryTypeEnumerationType">
      <restriction base="string">
        <enumeration value="Motorway"/>
        <enumeration value="RoadWithDualCarriageway"/>
        <enumeration value="RoadWithSingleCarriageway"/>
        <enumeration value="UrbanRoad"/>
        <enumeration value="ResidentialAreaRoad"/>
        <enumeration value="OtherRoad"/>
      </restriction>
    </simpleType>
    <simpleType name="RoadCategoryTypeOtherType">
      <restriction base="string">
        <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
    <simpleType name="FuzzyTimeType">
      <annotation>
        <documentation>Not precised described time.
[EuroRoadS]</documentation>
      </annotation>
      <union memberTypes="rst:FuzzyTimeEnumerationType rst:FuzzyTimeOtherType"/>
    </simpleType>
    <simpleType name="FuzzyTimeEnumerationType">
      <annotation>
        <documentation>Not precised described time.
[EuroRoadS]</documentation>
      </annotation>
      <restriction base="string">
        <enumeration value="External"/>
```

```
            <enumeration value="Dawn"/>
            <enumeration value="Dusk"/>
            <enumeration value="School"/>
            <enumeration value="Holiday"/>
            <enumeration value="Winter"/>
            <enumeration value="Spring"/>
            <enumeration value="Summer"/>
            <enumeration value="Autumn"/>
            <enumeration value="HighTide"/>
            <enumeration value="LowTide"/>
            <enumeration value="HighWater"/>
            <enumeration value="LowWater"/>
            <enumeration value="WetSeason"/>
            <enumeration value="DrySeason"/>
            <enumeration value="PeakHours"/>
            <enumeration value="OffPeakHours"/>
            <enumeration value="Day"/>
            <enumeration value="Night"/>
        </restriction>
    </simpleType>
    <simpleType name="FuzzyTimeOtherType">
        <restriction base="string">
            <pattern value="other: \w{2,}"/>
        </restriction>
    </simpleType>
    <element name="ValidityPeriod" type="rst:ValidityPeriodType" substitution-
Group="gml:AbstractObject">
        <annotation>
            <documentation>Definition of a validity period.
[TWG TN]</documentation>
        </annotation>
    </element>
    <complexType name="ValidityPeriodType">
        <sequence>
            <element name="timePeriodType" type="rst:TimePeriodTypeType" minOccurs="0"/>
            <element name="timePeriodValidity" type="rst:TimePeriodValidityType" minOccurs="0"/>
            <element name="description" type="string" minOccurs="0"/>
            <element name="year" type="rst:IntegerIntervalPropertyType" minOccurs="0"/>
            <element name="month" type="rst:IntegerIntervalPropertyType" minOccurs="0"/>
            <element name="week" type="rst:IntegerIntervalPropertyType" minOccurs="0"/>
            <element name="day" type="rst:IntegerIntervalPropertyType" minOccurs="0"/>
            <element name="weekdayType" type="rst:WeekdayTypeType" minOccurs="0"/>
            <element name="weekday" type="rst:IntegerIntervalPropertyType" minOccurs="0"/>
            <element name="time" type="rst:TimeIntervalType" minOccurs="0"/>
            <element name="fuzzyTimePeriod" type="rst:FuzzyTimePeriodPropertyType" minOc-
curs="0"/>
        </sequence>
    </complexType>
    <complexType name="ValidityPeriodPropertyType">
        <sequence>
            <element ref="rst:ValidityPeriod"/>
        </sequence>
    </complexType>
    <simpleType name="WeatherTypeType">
        <annotation>
            <documentation>Value list for weatherCondition.
```

```
[Euroroads]</documentation>
        </annotation>
        <union memberTypes="rst:WeatherTypeEnumerationType rst:WeatherTypeOtherType"/>
    </simpleType>
    <simpleType name="WeatherTypeEnumerationType">
        <annotation>
            <documentation>Value list for weatherCondition.
[Euroroads]</documentation>
        </annotation>
        <restriction base="string">
            <enumeration value="Fog"/>
            <enumeration value="Ice"/>
            <enumeration value="Rain"/>
            <enumeration value="AirPollution"/>
            <enumeration value="Snow"/>
            <enumeration value="Wind"/>
            <enumeration value="Wet"/>
        </restriction>
    </simpleType>
    <simpleType name="WeatherTypeOtherType">
        <restriction base="string">
            <pattern value="other: \w{2,}"/>
        </restriction>
    </simpleType>
    <simpleType name="TimePeriodValidityType">
        <annotation>
            <documentation>Defines if the time period is included or excluded.
[EuroRoadS]</documentation>
        </annotation>
        <restriction base="string">
            <enumeration value="IncludePeriod"/>
            <enumeration value="ExcludePeriod"/>
        </restriction>
    </simpleType>
    <element name="LaneCondition" type="rst:LaneConditionType" substitution-
Group="rst:Condition"/>
    <complexType name="LaneConditionType">
        <complexContent>
            <extension base="rst:ConditionType">
                <sequence>
                    <element name="startLane" type="integer"/>
                    <element name="laneExtension" type="integer"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="LaneConditionPropertyType">
        <sequence>
            <element ref="rst:LaneCondition"/>
        </sequence>
    </complexType>
    <simpleType name="TimePeriodTypeType">
        <annotation>
            <documentation>Defines if the time period occurs once or are repeated.
[EuroRoadS]</documentation>
        </annotation>
        <restriction base="string">
            <enumeration value="OccursOnce"/>
            <enumeration value="Yearly"/>
            <enumeration value="Monthly"/>
```

```
            <enumeration value="Weekly"/>
            <enumeration value="Daily"/>
        </restriction>
    </simpleType>
    <simpleType name="VehicleTypeType">
        <annotation>
            <documentation>A type of vehicle.
[TWG TN]</documentation>
        </annotation>
        <restriction base="string">
            <enumeration value="AllVehicle"/>
            <enumeration value="Bicycle"/>
            <enumeration value="CarWithTrailer"/>
            <enumeration value="DeliveryTruck"/>
            <enumeration value="EmergencyVehicle"/>
            <enumeration value="EmployeeVehicle"/>
            <enumeration value="FacilityVehicle"/>
            <enumeration value="FarmVehicle"/>
            <enumeration value="HighOccupancyVehicle"/>
            <enumeration value="LightRail"/>
            <enumeration value="MailVehicle"/>
            <enumeration value="MilitaryVehicle"/>
            <enumeration value="Moped"/>
            <enumeration value="Motorcycle"/>
            <enumeration value="PassengerCar"/>
            <enumeration value="Pedestrian"/>
            <enumeration value="PrivateBus"/>
            <enumeration value="PublicBus"/>
            <enumeration value="ResidentialVehicle"/>
            <enumeration value="SchoolBus"/>
            <enumeration value="SnowChainEquippedVehicle"/>
            <enumeration value="Tanker"/>
            <enumeration value="Taxi"/>
            <enumeration value="TransportTruck"/>
            <enumeration value="TrolleyBus"/>
            <enumeration value="VehicleForDisabledPerson"/>
            <enumeration value="VehicleWithSnowChainsOrSnowTyres"/>
        </restriction>
    </simpleType>
    <element name="ConditionSet" type="rst:ConditionSetType" substitutionGroup="rst:Condition"/>
    <complexType name="ConditionSetType">
        <complexContent>
            <extension base="rst:ConditionType">
                <sequence>
                    <element name="conditions" type="rst:ConditionPropertyType" maxOc-
curs="unbounded"/>
                    <element name="operator" type="rst:ConditionOperatorType"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="ConditionSetPropertyType">
        <sequence>
            <element ref="rst:ConditionSet"/>
        </sequence>
    </complexType>
```

```
    <element name="FuzzyTimePeriod" type="rst:FuzzyTimePeriodType" substitution-
Group="gml:AbstractObject">
      <annotation>
        <documentation>Time described with a fuzzy time.
[EuroRoadS]</documentation>
      </annotation>
    </element>
    <complexType name="FuzzyTimePeriodType">
      <sequence>
        <element name="beginOrDuration" type="rst:FuzzyTimeType"/>
        <element name="endOrDuration" type="rst:FuzzyTimeType" minOccurs="0"/>
      </sequence>
    </complexType>
    <complexType name="FuzzyTimePeriodPropertyType">
      <sequence>
        <element ref="rst:FuzzyTimePeriod"/>
      </sequence>
    </complexType>
    <element name="WeatherCondition" type="rst:WeatherConditionType" substitution-
Group="rst:Condition"/>
    <complexType name="WeatherConditionType">
      <complexContent>
        <extension base="rst:ConditionType">
          <sequence>
            <element name="weatherType" type="rst:WeatherTypeType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="WeatherConditionPropertyType">
      <sequence>
        <element ref="rst:WeatherCondition"/>
      </sequence>
    </complexType>
    <element name="TimeCondition" type="rst:TimeConditionType" substitution-
Group="rst:Condition"/>
    <complexType name="TimeConditionType">
      <complexContent>
        <extension base="rst:ConditionType">
          <sequence>
            <element name="validityPeriod" type="rst:ValidityPeriodPropertyType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="TimeConditionPropertyType">
      <sequence>
        <element ref="rst:TimeCondition"/>
      </sequence>
    </complexType>
    <simpleType name="WeekdayTypeType">
      <annotation>
        <documentation>Defines where the week number is counted from.
[EuroRoadS]</documentation>
      </annotation>
      <restriction base="string">
        <enumeration value="Absolute"/>
        <enumeration value="FromEndOfMonth"/>
        <enumeration value="FromStartOfMonth"/>
      </restriction>
```

```
    </simpleType>
    <simpleType name="AreaTypeType">
      <annotation>
          <documentation>Specifies a reason for the existance of the safety fea-
ture</documentation>
      </annotation>
      <union memberTypes="rst:AreaTypeEnumerationType rst:AreaTypeOtherType"/>
    </simpleType>
    <simpleType name="AreaTypeEnumerationType">
      <annotation>
          <documentation>Specifies a reason for the existance of the safety fea-
ture</documentation>
      </annotation>
      <restriction base="string">
          <enumeration value="InNationalPark"/>
          <enumeration value="InsideCities"/>
          <enumeration value="NearRailroadCrossing"/>
          <enumeration value="NearSchool"/>
          <enumeration value="OutsideCities"/>
          <enumeration value="TrafficCalmingArea"/>
      </restriction>
    </simpleType>
    <simpleType name="AreaTypeOtherType">
      <restriction base="string">
          <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
    <simpleType name="LoadTypeType">
      <union memberTypes="rst:LoadTypeEnumerationType rst:LoadTypeOtherType"/>
    </simpleType>
    <simpleType name="LoadTypeEnumerationType">
      <restriction base="string">
          <enumeration value="ExplosiveLoad"/>
          <enumeration value="WaterPollutingLoad"/>
          <enumeration value="OtherDangerousLoad"/>
      </restriction>
    </simpleType>
    <simpleType name="LoadTypeOtherType">
      <restriction base="string">
          <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
    <element name="VehicleCondition" type="rst:VehicleConditionType" substitution-
Group="rst:Condition"/>
    <complexType name="VehicleConditionType">
      <complexContent>
          <extension base="rst:ConditionType">
            <sequence>
                <element name="vehicleType" type="rst:VehicleTypeType"/>
            </sequence>
          </extension>
      </complexContent>
    </complexType>
    <complexType name="VehicleConditionPropertyType">
      <sequence>
          <element ref="rst:VehicleCondition"/>
```

```
      </sequence>
    </complexType>
    <element name="TimeInterval" type="rst:TimeIntervalType" substitution-
Group="gml:AbstractObject"/>
    <complexType name="TimeIntervalType">
      <sequence>
        <element name="begin" type="time"/>
        <element name="lengthSeconds" type="integer" minOccurs="0"/>
      </sequence>
    </complexType>
    <complexType name="TimeIntervalPropertyType">
      <sequence>
        <element ref="rst:TimeInterval"/>
      </sequence>
    </complexType>
</schema>
```

### *10.1.5. LocationReferencing.xsd*

```
<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <!--XML Schema document created by ShapeChange-->
    <element name="LocationReference" type="rst:LocationReferenceType" abstract="true" substitu-
tionGroup="gml:AbstractGML"/>
    <complexType name="LocationReferenceType" abstract="true">
      <complexContent>
        <extension base="gml:AbstractGMLType">
          <sequence/>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="LocationReferencePropertyType">
      <sequence minOccurs="0">
        <element ref="rst:LocationReference"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="DirectLocationReference" type="rst:DirectLocationReferenceType" ab-
stract="true" substitutionGroup="rst:LocationReference"/>
    <complexType name="DirectLocationReferenceType" abstract="true">
      <complexContent>
        <extension base="rst:LocationReferenceType">
          <sequence/>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="DirectLocationReferencePropertyType">
      <sequence minOccurs="0">
        <element ref="rst:DirectLocationReference"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
```

```
    <element name="IndirectLocationReference" type="rst:IndirectLocationReferenceType" ab-
stract="true" substitutionGroup="rst:LocationReference"/>
    <complexType name="IndirectLocationReferenceType" abstract="true">
      <complexContent>
        <extension base="rst:LocationReferenceType">
          <sequence/>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="IndirectLocationReferencePropertyType">
      <sequence minOccurs="0">
        <element ref="rst:IndirectLocationReference"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="AgoraLocationString" type="rst:AgoraLocationStringType" abstract="false" subs-
titutionGroup="rst:DirectLocationReference"/>
    <complexType name="AgoraLocationStringType" abstract="false">
      <complexContent>
        <extension base="rst:DirectLocationReferenceType">
          <sequence>
            <element name="base64String" type="string"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="AgoraLocationStringPropertyType">
      <sequence minOccurs="0">
        <element ref="rst:AgoraLocationString"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
</schema>
```

### *10.1.6. Update.xsd*

```
<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <!--XML Schema document created by ShapeChange-->
    <simpleType name="UpdateTypeType">
      <restriction base="string">
        <enumeration value="Add"/>
        <enumeration value="Modify"/>
        <enumeration value="Remove"/>
      </restriction>
    </simpleType>
    <element name="UpdateInfo" type="rst:UpdateInfoType" substitution-
Group="gml:AbstractObject"/>
    <complexType name="UpdateInfoType">
      <sequence>
```

```
            <element name="type" type="rst:UpdateTypeType"/>
        </sequence>
    </complexType>
    <complexType name="UpdateInfoPropertyType">
        <sequence>
            <element ref="rst:UpdateInfo"/>
        </sequence>
    </complexType>
</schema>
```

### 10.1.7. Feedback.xsd

```
<?xml version="1.0" encoding="windows-1252"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:gml="http://www.opengis.net/gml"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst" targetNames-
pace="http://www.ertico.com/en/subprojects/rosatte/rst" elementFormDefault="qualified" ver-
sion="2009-05-20">
    <include schemaLocation="SafetyFeatures.xsd"/>
    <import namespace="http://www.opengis.net/gml" schemaLoca-
tion="http://www.isotc211.org/schemas/2005/gml/gml.xsd"/>
    <!--XML Schema document created by ShapeChange-->
    <simpleType name="SafetyFeatureEventCodeType">
        <union memberTypes="rst:SafetyFeatureEventCodeEnumerationType
rst:SafetyFeatureEventCodeOtherType"/>
    </simpleType>
    <simpleType name="SafetyFeatureEventCodeEnumerationType">
        <restriction base="string">
            <enumeration value="DecodeLocationError"/>
            <enumeration value="DecodeLocationErrorGeometryMismatch"/>
            <enumeration value="DecodeLocationRoadDescriptorMismatch"/>
            <enumeration value="IllegalPropertyValue"/>
        </restriction>
    </simpleType>
    <simpleType name="SafetyFeatureEventCodeOtherType">
        <restriction base="string">
            <pattern value="other: \w{2,}"/>
        </restriction>
    </simpleType>
    <element name="FreeTextEvent" type="rst:FreeTextEventType" substitution-
Group="rst:FeedbackLogEvent"/>
    <complexType name="FreeTextEventType">
        <complexContent>
            <extension base="rst:FeedbackLogEventType">
                <sequence>
                    <element name="text" type="string"/>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="FreeTextEventPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:FreeTextEvent"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="FeedbackInformation" type="rst:FeedbackInformationType" substitution-
Group="gml:AbstractGML"/>
    <complexType name="FeedbackInformationType">
        <complexContent>
```

```
            <extension base="gml:AbstractGMLType">
              <sequence>
                  <element name="log" type="rst:FeedbackLogPropertyType"/>
                  <element name="datasetIdentifier" type="string"/>
                  <element name="processed" type="dateTime"/>
              </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="FeedbackInformationPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:FeedbackInformation"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="GeneralFeedbackEvent" type="rst:GeneralFeedbackEventType" substitution-
Group="rst:FeedbackLogEvent"/>
    <complexType name="GeneralFeedbackEventType">
        <complexContent>
            <extension base="rst:FeedbackLogEventType">
              <sequence>
                  <element name="eventCode" type="rst:GeneralFeedbackEventCodeType"/>
              </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="GeneralFeedbackEventPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:GeneralFeedbackEvent"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <element name="SafetyFeatureEvent" type="rst:SafetyFeatureEventType" substitution-
Group="rst:FeedbackLogEvent"/>
    <complexType name="SafetyFeatureEventType">
        <complexContent>
            <extension base="rst:FeedbackLogEventType">
              <sequence>
                  <element name="safetyFeatureId" type="rst:SafetyFeatureIdPropertyType"/>
                  <element name="safetyFeaturePropertyType"
type="rst:SafetyFeaturePropertyTypeCodeType" minOccurs="0"/>
                  <element name="eventCode" type="rst:SafetyFeatureEventCodeType"/>
              </sequence>
            </extension>
        </complexContent>
    </complexType>
    <complexType name="SafetyFeatureEventPropertyType">
        <sequence minOccurs="0">
            <element ref="rst:SafetyFeatureEvent"/>
        </sequence>
        <attributeGroup ref="gml:AssociationAttributeGroup"/>
        <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
```

```
    <element name="FeedbackLogEvent" type="rst:FeedbackLogEventType" abstract="true" substitu-
tionGroup="gml:AbstractGML"/>
    <complexType name="FeedbackLogEventType" abstract="true">
      <complexContent>
        <extension base="gml:AbstractGMLType">
          <sequence/>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="FeedbackLogEventPropertyType">
      <sequence minOccurs="0">
        <element ref="rst:FeedbackLogEvent"/>
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
    <simpleType name="GeneralFeedbackEventCodeType">
      <union memberTypes="rst:GeneralFeedbackEventCodeEnumerationType
rst:GeneralFeedbackEventCodeOtherType"/>
    </simpleType>
    <simpleType name="GeneralFeedbackEventCodeEnumerationType">
      <restriction base="string">
        <enumeration value="SchemaVersionMismatch"/>
      </restriction>
    </simpleType>
    <simpleType name="GeneralFeedbackEventCodeOtherType">
      <restriction base="string">
        <pattern value="other: \w{2,}"/>
      </restriction>
    </simpleType>
    <element name="SummaryInfo" type="rst:SummaryInfoType" substitution-
Group="gml:AbstractObject"/>
    <complexType name="SummaryInfoType">
      <sequence>
        <element name="nrSuccess" type="integer"/>
        <element name="nrFail" type="integer"/>
      </sequence>
    </complexType>
    <complexType name="SummaryInfoPropertyType">
      <sequence>
        <element ref="rst:SummaryInfo"/>
      </sequence>
    </complexType>
    <element name="FeedbackLog" type="rst:FeedbackLogType" substitution-
Group="gml:AbstractGML"/>
    <complexType name="FeedbackLogType">
      <complexContent>
        <extension base="gml:AbstractGMLType">
          <sequence>
            <element name="events" type="rst:FeedbackLogEventPropertyType" minOccurs="0"
maxOccurs="unbounded"/>
            <element name="summary" type="rst:SummaryInfoPropertyType"/>
          </sequence>
        </extension>
      </complexContent>
    </complexType>
    <complexType name="FeedbackLogPropertyType">
      <sequence minOccurs="0">
        <element ref="rst:FeedbackLog"/>
      </sequence>
```

```
      <attributeGroup ref="gml:AssociationAttributeGroup"/>
      <attributeGroup ref="gml:OwnershipAttributeGroup"/>
    </complexType>
</schema>
```

## 10.2. Examples of gml files

To help understanding the xsd schemas in the previous section a number of concrete examples of data complying with the schemas are added. More examples shall be added in the final version of this document.

### 10.2.1. Example SafetyFeature

```
<?xml version="1.0" encoding="UTF-8"?>
<rst:ROSATTESafetyFeatureDataset gml:id="i0"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst"
xmlns:gml="http://www.opengis.net/gml" xmlns:gmd="http://www.isotc211.org/2005/gmd"
xmlns:gco="http://www.isotc211.org/2005/gco" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.ertico.com/en/subprojects/rosatte/rst
.\\ROSATTE.xsd">
    <gml:featureMember>
      <rst:GenericSafetyFeature gml:id="i1">
          <rst:id>
            <rst:SafetyFeatureId>
                <rst:providerId>abc123</rst:providerId>
                <rst:id>1</rst:id>
            </rst:SafetyFeatureId>
          </rst:id>
          <rst:locationReference><rst:AgoraLocationString
gml:id="i2"><rst:base64String>abcd</rst:base64String></rst:AgoraLocationString></rst:locationRefer
ence>
          <rst:validFrom>2009-05-20</rst:validFrom>
          <rst:updateInfo>
            <rst:UpdateInfo>
                <rst:type>Add</rst:type>
            </rst:UpdateInfo>
          </rst:updateInfo>
          <rst:source>Regulation</rst:source>
          <rst:type>SpeedLimit</rst:type>
          <rst:properties>
            <rst:SafetyFeaturePropertyValue>
                <rst:type>MaximumSpeedLimit</rst:type>
                <rst:propertyValue><gml:measure
uom="kmph">100</gml:measure></rst:propertyValue>
            </rst:SafetyFeaturePropertyValue>
          </rst:properties>
          <rst:condition>
            <rst:ConditionSet>
                <rst:conditions>
                  <rst:TimeCondition>
                      <rst:validityPeriod>
                          <rst:ValidityPeriod>
                            <rst:weekday>
                                <rst:IntegerInterval>
                                <rst:start>1</rst:start>
```

```
                                    <rst:length>5</rst:length>
                                    </rst:IntegerInterval>
                                </rst:weekday>

    <rst:time><rst:begin>09:00:00</rst:begin><rst:lengthSeconds>32400</rst:lengthSeconds></rst:ti
me>
                                </rst:ValidityPeriod>
                            </rst:validityPeriod>
                        </rst:TimeCondition>
                    </rst:conditions>
                    <rst:conditions>
                        <rst:VehicleCondition>
                            <rst:negate>true</rst:negate>
                            <rst:vehicleType>SchoolBus</rst:vehicleType>
                        </rst:VehicleCondition>
                    </rst:conditions>
                    <rst:operator>AND</rst:operator>
                </rst:ConditionSet>
            </rst:condition>
        </rst:GenericSafetyFeature>
    </gml:featureMember>
    <rst:metadata>
        <gmd:contact><gmd:CI_ResponsibleParty>
            <gmd:organisationName><gco:CharacterString>Triona
AB</gco:CharacterString></gmd:organisationName>
            <gmd:role><gmd:CI_RoleCode codeListValue="Supplier" codeL-
ist="http://www.ertico.com/en/subprojects/rosatte/rst/RoleCodes.xml"></gmd:CI_RoleCode></gm
d:role>
        </gmd:CI_ResponsibleParty></gmd:contact>
        <gmd:dateStamp><gco:DateTime>2009-05-22T11:00:00</gco:DateTime></gmd:dateStamp>
        <gmd:identificationInfo><gmd:MD_DataIdentification>
            <gmd:citation><gmd:CI_Citation>
                <gmd:title><gco:CharacterString>{DA9C0AB3-C134-4DF9-BD53-
9B5A4E22E95D}/{BFC5F8A7-9B74-4032-BA45-DA4511CC1143}</gco:CharacterString></gmd:title>
                <gmd:date><gmd:CI_Date>
                    <gmd:date><gco:DateTime>2009-05-22T11:00:00</gco:DateTime></gmd:date>
                    <gmd:dateType><gmd:CI_DateTypeCode codeListValue="creation" codeL-
ist="http://www.ertico.com/en/subprojects/rosatte/rst/DateTypeCodes.xml"></gmd:CI_DateTypeC
ode></gmd:dateType>
                </gmd:CI_Date></gmd:date>
            </gmd:CI_Citation></gmd:citation>
            <gmd:abstract><gco:CharacterString>A first example of a ROSATTE data-
set</gco:CharacterString></gmd:abstract>
            <gmd:language><gco:CharacterString>eng</gco:CharacterString></gmd:language>
        </gmd:MD_DataIdentification></gmd:identificationInfo>
    </rst:metadata>
    <rst:type>Snapshot</rst:type>
</rst:ROSATTESafetyFeatureDataset>
```

### *10.2.2. Example Feedback*

```
<?xml version="1.0" encoding="UTF-8"?>
<rst:ROSATTEFeedbackDataset gml:id="i0"
xmlns:rst="http://www.ertico.com/en/subprojects/rosatte/rst"
xmlns:gml="http://www.opengis.net/gml" xmlns:gmd="http://www.isotc211.org/2005/gmd"
xmlns:gco="http://www.isotc211.org/2005/gco" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.ertico.com/en/subprojects/rosatte/rst
.\\ROSATTE.xsd">
    <rst:feedbackInfo>
        <rst:FeedbackInformation gml:id="i1">
```

```
        <rst:log>
          <rst:FeedbackLog gml:id="i2">
            <rst:events>
              <rst:FreeTextEvent gml:id="i3">
                <rst:text>A free text error</rst:text>
              </rst:FreeTextEvent>
            </rst:events>
            <rst:events>
              <rst:GeneralFeedbackEvent gml:id="i4">
                <rst:eventCode>SchemaVersionMismatch</rst:eventCode>
              </rst:GeneralFeedbackEvent>
            </rst:events>
            <rst:events>
              <rst:SafetyFeatureEvent gml:id="i5">
                <rst:safetyFeatureId>
                  <rst:SafetyFeatureId>
                    <rst:providerId>abc123</rst:providerId>
                    <rst:id>1</rst:id>
                  </rst:SafetyFeatureId>
                </rst:safetyFeatureId>
                <rst:eventCode>DecodeLocationError</rst:eventCode>
              </rst:SafetyFeatureEvent>
            </rst:events>
            <rst:summary>
              <rst:SummaryInfo>
                <rst:nrSuccess>123</rst:nrSuccess>
                <rst:nrFail>2</rst:nrFail>
              </rst:SummaryInfo>
            </rst:summary>
          </rst:FeedbackLog>
        </rst:log>
        <rst:datasetIdentifier>{DA9C0AB3-C134-4DF9-BD53-9B5A4E22E95D}/{BFC5F8A7-9B74-4032-
BA45-DA4511CC1143}</rst:datasetIdentifier>
        <rst:processed>2009-05-29T15:00:00</rst:processed>
      </rst:FeedbackInformation>
    </rst:feedbackInfo>
</rst:ROSATTEFeedbackDataset>
```