UTMC Data Exchange

Product Specification

Issue 16

Contents

Minimum distribution

Name	Company	Requirements originator
Richard Thurbin	Cloud Amber	Technical Director
Mike Wells	Cloud Amber	Operations Director

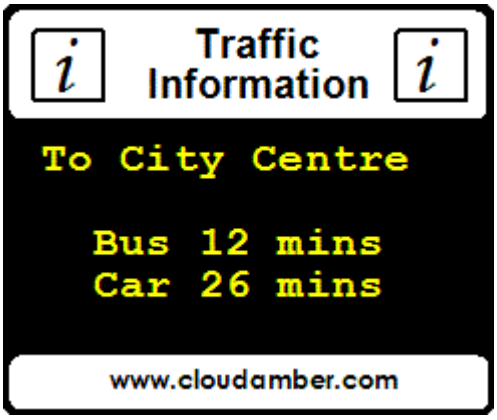
1 Introduction

1.1 Cloud Amber

Cloud Amber approaches UTMC from a different perspective from the traditional companies. We find Urban Traffic Management Control to be constricting and anachronistic where Traffic Managers are obliged to take account of Walking, Cycling and Public Transport. Instead, we prefer to interpret the acronym as Unified Transport Management Control.

Traditionally, the movement of people and goods has been divided into two distinct spheres with their own monitoring and control systems: Public Transport and Traffic. Cloud Amber tries to provide a holistic view of the Transport network with products designed to bridge this divide which provide.

In managing their networks through fine tuning Traffic Signal timings, there is an implicit acceptance by Traffic Managers that the number of vehicles on the network is fixed and does not recognise that one of the ways in which the network can be managed more effectively is by managing drivers' behaviour and either suggesting that they postpone their journey to a time when the network is less congested or by suggesting a public transport alternative.



Equally, Public Transport Managers have spent millions of pounds on improving bus services through measures such as Real Time Passenger information with the intention of 'encouraging modal shift' and yet this information has largely been aimed at existing public transport passengers. For modal shift to take place, Cloud Amber believes that Public Transport Managers should in fact be addressing their information to a totally different constituency- the car users.

It has almost been the case that Public and Private Transport modes are mutually exclusive. For example, the Transport Direct website does not suggest that it would be possible to drive the first 10 miles to a Railway Station and then take the train, instead an absurd 3 changes of bus is proposed before the train journey with the outcome that anyone with a car would just drive the entire journey.

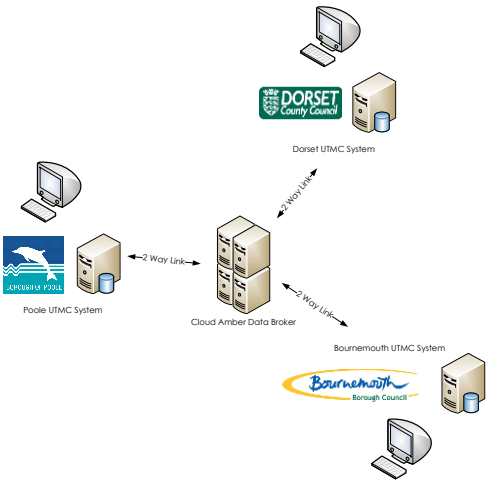
Cloud Amber believes drivers would be more likely to undertake the model shift if public and private transport were available in one place with a clear, unbiased comparison between the two and an acceptance that it is possible to combine modes.

In addition, Cloud Amber is aware that the Counties are not islands, travellers cross boundaries and have no interest in researching journey times on a multitude of sites. Traffic Managers also need to be aware of what the traffic conditions are over the border outside their jurisdiction.

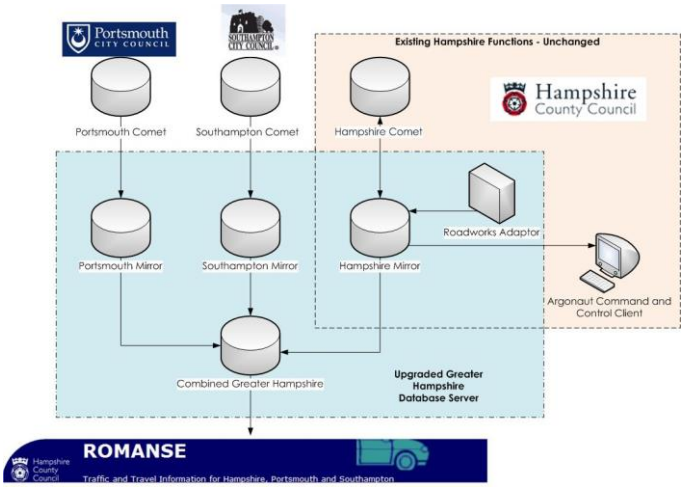
For Traffic Managers, Cloud Amber has extended the UTMC standard so that systems may be linked and could, if required be jointly controlled. Examples of this would be:

- Dorset/ Poole/ Bournemouth and Hampshire and the HA
- Bedford Borough/ Central Bedfordshire and Cambridgeshire and the HA
- Oxfordshire/ Warwickshire/ Reading and the HA

Poole, Bournemouth & Dorset



Hampshire, Portsmouth & Southampton



Cloud Amber's core belief in the availability of information to each, according to their need, the democratisation of information guarantees a user focused approach to design work which ensures that the enhanced data provided by highly sophisticated modelling and interrogative software can be easily reached via simple user friendly front ends.

With unparalleled experience in both the Public and Private Transport Industries, Cloud Amber is uniquely placed with the skills and knowledge and experience of the wide range of standards and protocols, to combine all the various data feeds. A large group of partners in fields such as UTC, mapping, car parking, journey planning and telephony as well as specialist software developers has enabled Cloud Amber to integrate the most innovative solutions into our product.

The company is committed to minimising the environmental impact of travel whilst enabling the unhindered movement of people necessary for a growing economy. This requires the use of modern information technology and communications methods - Intelligent Transport Systems (ITS) - to allow best use to be made of networks both for

private and public transport, facilitate links between transport modes and mitigate adverse impacts.

1.1.1 Voyager

The Cloud Amber Voyager product provides a whole series of dissemination output channels for transportation information. Built using the latest Microsoft technology, the Voyager outputs have been designed to make accessing using, reading and surfing as simple and easy as possible. The current dissemination channels available include web, mobile, SMS, DiTV, mp3 download, kiosks, broadcast screens and links to satellite navigation systems. For more information see

http://voyager.cloudamber.com

1.1.2 Argonaut

The Cloud Amber Argonaut Common Database is a UTMC compliant database which provides an open standard data store to collect and disseminate traffic and transport information from existing UTC sub systems. It also provides Traffic Managers with the ability to develop better, integrated traffic and transport control strategies in line with current national and local transport policies. For more information see

http://argonaut.cloudamber.com

1.2 Data Broker

Multiple disparate data sources already exist to describe the state of the transport network. Cloud Amber combines these into a single source- the Argonaut UTMC Common Database via our Data Broker to enable Traffic Managers to monitor, interrogate, model, and intervene to maximise the use of a limited road network, thereby managing the environmental effects of travel as well as reducing congestion.

Our unique data broker combines all UTMC and Public Transport information in to a common data architecture. Cloud Amber may export all data in Datex2 compliant format and supports FTP, HTTP GET and SOAP requests through our Data Broker. Put simply, through a web service interface, a single entity can gain access to all available traffic and public transport information without the hassle of endless single system connections. We have learnt the hard way through our work in South Yorkshire, Thames Valley and Dorset/Poole/Bournemouth how UTMC doesn't extend beyond the confines of an authorities boundary!

Features

- Single interface for Traffic and Public Transport
- Easy to integrate, XML Web Service interface
- Fully supported with example Java and .Net applications
- Supports 2 way data transfer
- Integrates with over 30 separate data sources throughout England and Wales, including Cardiff County Council, Somerset, Worcestershire County Council and both Cheshire authorities.
- Fully integrates with all other UTMC system providers
- Supports UTMC, DatexII, Freeflow& more
- Centralised data source
- Publish and subscribe supported
- Request and response supported
- Compatible with UK & European standards
- Integrates with NaPTAN, NPTEG & other associated UK standards
- Integrates with displays, kiosks, SMS, Traveline and many other systems

Benefits

- Guaranteed lower capital and maintenance
- Obviates the need for numerous connections
- Ensures competitive prices for displays and other equipment
- Wider market and product set for available purchase
- Single point of contact

Link to example live public portal for the Thames Valley being fed directly from the combined data broker: http://www.thamesvalleyvoyager.com

Access to the above site is restricted by a username and password. Therefore, please contact James Hewetson at Cloud Amber for user credentials.

1.3 Data Standards

1.3.1 UTMC

Launched in 1997, the UTMC programme was the UK Department for Transport (DfT) main initiative for the development of a more open approach to Intelligent Transport Systems (ITS) in urban areas. During the first three years, a number of research projects were undertaken to establish and validate an approach based on modular systems and open standards. These have contributed to the UTMC Technical Specifications, which define UTMC standards.

In January 2001, the programme embarked on its demonstrator phase to consolidate the results of the earlier research. Preston, Reading, Stratford-upon-Avon and York have implemented, in a pragmatic way, full scale demonstrator projects based on the UTMC approach.

More information on the Research and Demonstrator phases of the UTMC Initiative can be found at: www.utmc.gov.uk.

Early in 2003, the UTMC Development Group (UDG) of stakeholders, consisting of local authorities and suppliers was set up, with the support of the DfT, to oversee the future development of UTMC.

1.3.2 Datex II

Delivering European Transport Policy in line with the ITS Action Plan of the European Commission requires co-ordination of traffic management and development of seamless pan European services.

With the aim to support sustainable mobility in Europe, the European Commission has been supporting the development of information exchange mainly between the actors of the road traffic management domain for a number of years. In the road sector, the DATEX standard was developed for information exchange between traffic management centres, traffic information centres and service providers and constitutes the reference for applications that have been developed in the last 10 years.

The second generation DATEX II specification now also pushes the door wide open for all actors in the traffic and travel information sector.

Much investment has been made in Europe, both in traffic control and information centres over the last decade and also in a quantum shift in the monitoring of the trans-European transport network (TEN-T). This is in line with delivering the objectives of the EasyWay programme for safer roads, reduced congestion and a better environment. Collecting information is only part of the story – to make the most of the investment data needs to be exchanged both with other centres and, in a more recent development, with those developing pan-European services provided directly to road users.

DATEX was originally designed and developed as a traffic and travel data exchange mechanism by a European task force set up to standardise the interface between traffic control and information centres.

With the new generation DATEX II it has become the reference for all applications requiring access to dynamic traffic and travel related information in Europe.

http://www.datex2.eu/

2 Architecture

2.1 Concept

Cloud Amber has designed and developed our system software using the highly flexible and robust 3-tier architecture. This was designed for the specific purpose of integrating as many systems into UTMC as simply and effortlessly as possible, without involving either large amounts of development time or overhead.

2.1.1 What is a Layer?

A layer is a reusable portion of code that performs a specific function. A layer is usually setup as a project that represents this specific function. This specific layer is in charge of working with other layers to perform some specific goal. In an application where the presentation layer needs to extract information from a backend database, the presentation would utilise a series of layers to retrieve the data, rather than having the database calls embedded directly within itself.

2.1.2 The Data Layer

The key component to most applications is the data. The data has to be served to the presentation layer somehow. The data layer is a separate component (often setup as a separate single or group of projects in a .NET solution), whose sole purpose is to serve up the data from the database and return it to the caller. Through this approach, data can be logically reused, meaning that a portion of an application reusing the same query can make a call to one data layer method, instead of embedding the query multiple times. This is generally more maintainable.

2.1.3 Business Layer

Though a web site could talk to the data access layer directly, it usually goes through another layer called the business layer. The business layer is vital in that it validates the input conditions before calling a method from the data layer. This ensures the data input is correct before proceeding, and can often ensure that the outputs are correct as well. This validation of input is called business rules, meaning the rules that the business layer uses to make "judgments" about the data.

One of the best reasons for reusing logic is that applications that start off small usually grow in functionality. For instance, a company begins to develop a web site, and as they realize their business needs, they later decide to add a smart client application and windows service to supplement the web site. The business layer helps move logic to a central layer for "maximum reusability."

2.1.4 Presentation Layer

The web site or windows forms application (the UI for the project) is called the presentation layer. The presentation layer is the most important layer simply because it's the one that everyone sees and uses. Cloud Amber removes as much business logic out of the UI and into the business layer. This usually involves more code, but in my mind, the excess time

pays off in the end with a clean and crisp design, allowing a greater range of output formats and devices.

2.1.5 Why Separating Logic Is Useful

The biggest reason is reuse: logic placed in a business layer increases the reusability of an application. As applications grow, applications often grow into other realms. Applications may start out as a web application, but some of the functionality may later be moved to a smart client application. Portions of an application may be split between a web site and a web or windows service that runs on a server. In addition, keeping logic helps aid in developing a good design (sometimes code can get sloppier in the UI).

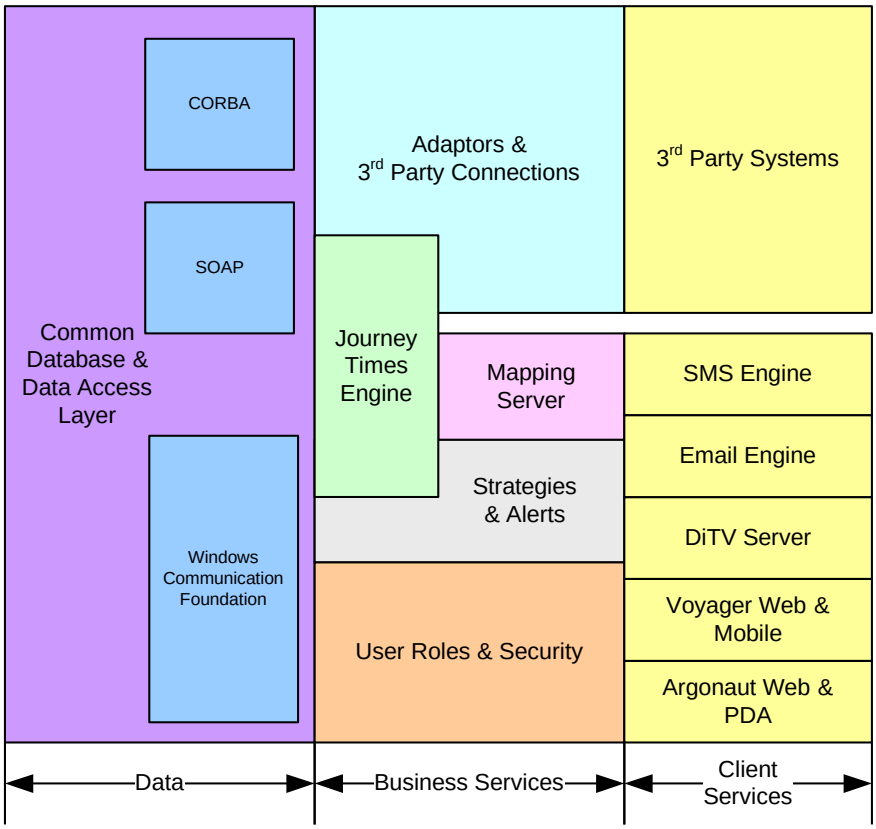
However, there are some caveats to this: it takes a little longer to develop applications when most of the logic resides in the business layer. The reason is this often involves creating several sets of objects (data layer and access code, plus business objects) rather than embedding it in the application. The extra time that it takes to do this can be a turnoff for some managers and project leads, especially because it often requires you to be knowledgeable about object-oriented programming, more than most people are comfortable with.

Although embedding code in the UI is easier, in most cases Cloud Amber doesn't believe it's the best approach. A layered approach is often a better approach because it pays dividends down the road. This is because as more and more code is developed, the following happens:

- Code is copied and pasted frequently, or code is reused in classes that could easily be moved to a business layer.
- Code that is very similar is often copied and pasted with slight modification, making duplication harder to track down.
- It's harder to maintain; even though applications with business objects are larger applications, they usually are structured better.
- Code is harder to unit test, if unit testing is available at all. Web applications and windows forms projects are hard to use unit testing with.
- A good architecture is often harder to implement, but is easier to maintain because it often reduces the volume of code. This means that hours spent supporting an application are reduced.

2.2 System Implementation

Cloud Amber's implementation of the 3 tier architecture is shown below. Cloud Amber shall provide full integration with data access and business services layers, allowing integration with other system, including COBS via the SSL library.



Third paries interface in to the Cloud Amber Data Broker system using the business services layer and act as a client service.

3 UTMC Traffic Data Interface Using Datex II

3.1 Interface Specification

There are a number of ways systems can access Datex II data. They are via FTP, HTTP GET. Cloud Amber is currently developing SOAP Web Services and Windows Communication Framework interfaces but are currently not avaliable for use.

In order to access the data, systems will require a user name and password. The credentials are supplied by the operations team.

This document refers to Datex II v1.0.

3.1.1 FTP

Every 5 minutes, our Data Broker will update a file on to a FTP server containing all of the information inside the common database in Datex II format.

The file name remains constant and the name of the file to be downloaded will be advised by the operations team.

Both active and passive connections are supported.

In order to access the data, systems will need to log in to the FTP server. Anonymous connections are refused.

3.1.2 HTTP GET

Systems may request Datex II data using a HTTP get request.

The request URL will differ from implementation to implementation and will be supplied by the operations team.

The get request has the following parameters:

- UserName
- Password

Upon receiving a get request with invalid credentials, a ***401 Unauthorized*** status code will be returned with no data.

Upon receiving a authorised get request, the system shall transmit a cached copy of the data available on the FTP server.

Users who request more than one GET request within a five minute period may get returned a ***304 Not Modified*** status code with no data.

The filename remains constant.

3.2 Data Map

In order to publish Datex II data stored in UTMC format a number of mappings has been done. These are in accordance with the ITS Metadata Registry. The Registry is a repository of data definitions and data models, with an associated supporting process for improvement of quality and for harmonisation across different systems. The registry aims to cut across work in isolated "silos" and avoid re-invention and duplication of effort.

For more information visit: http://www.itsregistry.org.uk

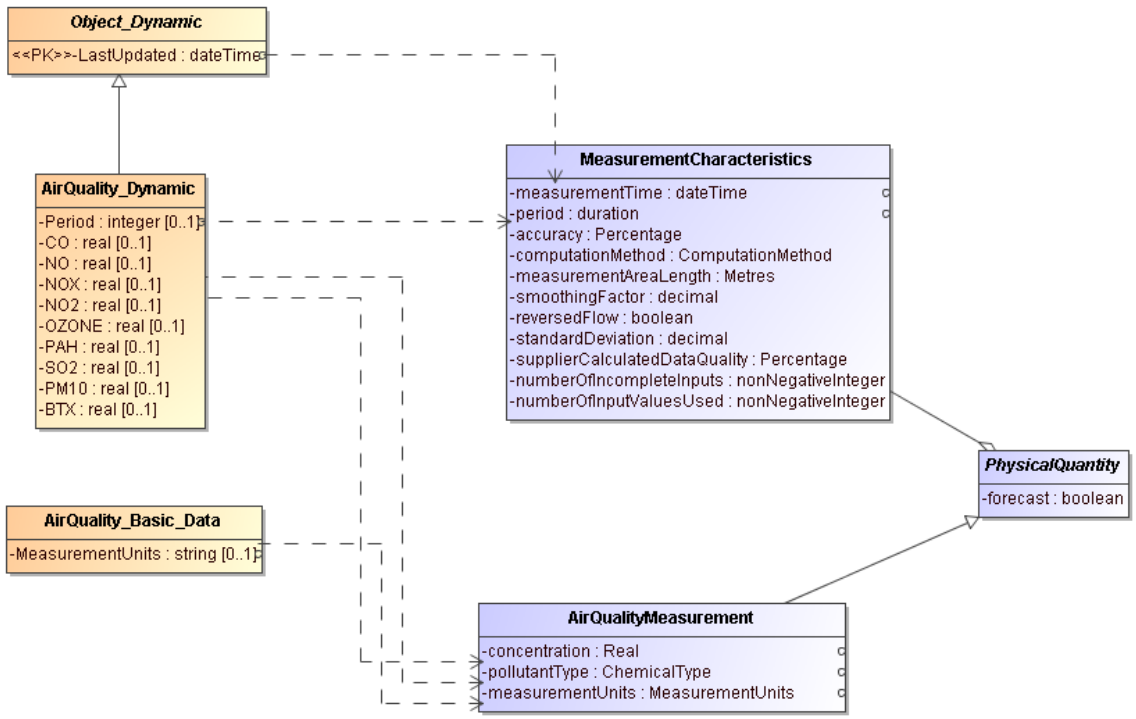
The following details diagrams which show the map of data between the UTMC common database in orange and the Datex II specification in purple.

For more information on what the data entities and attributes represent, please refer to their respective standards.

3.2.1 Air Quality

The scope is limited to the publication of air quality information and does not include control of air quality.

Entity and Attribute



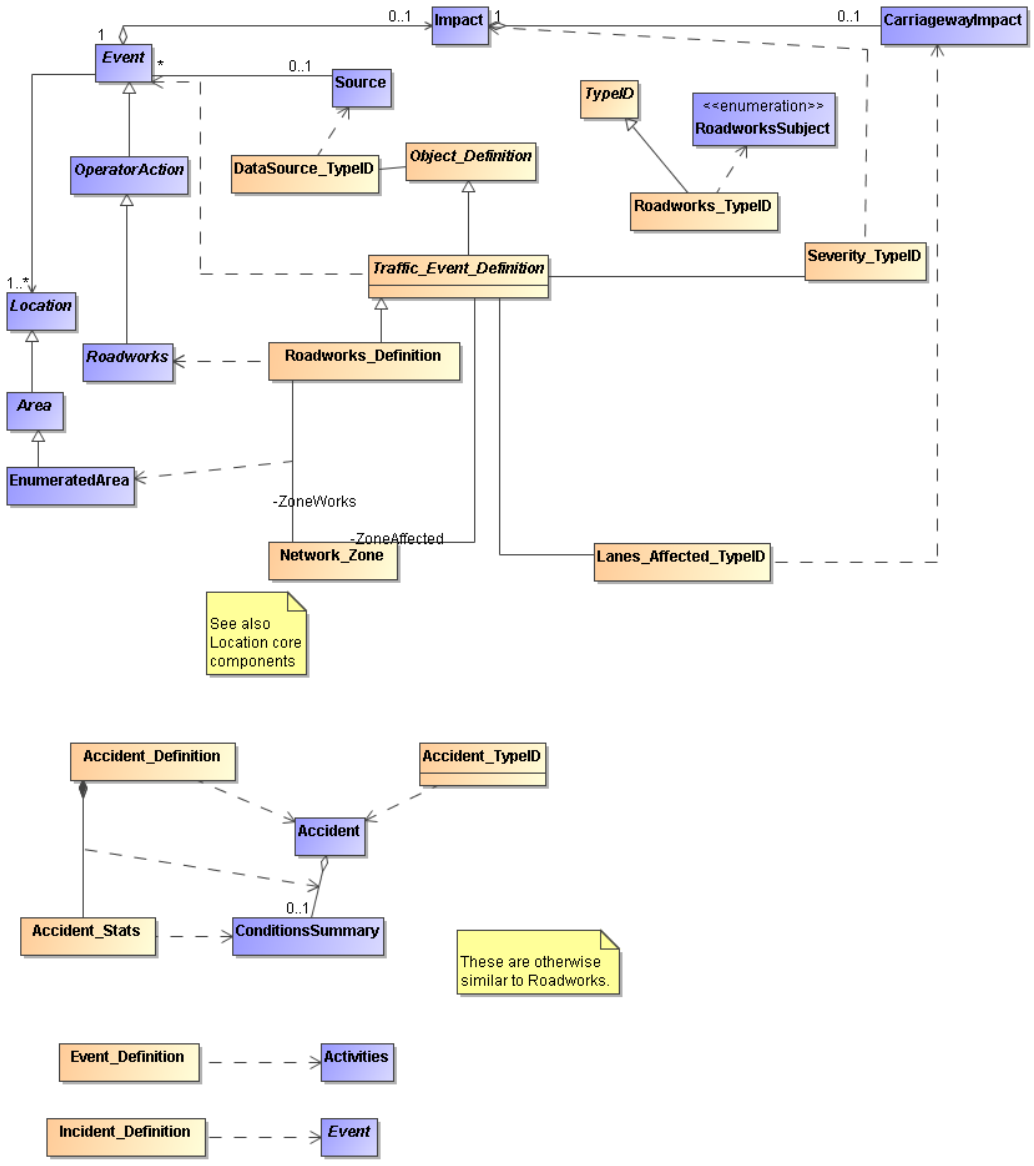
3.2.2 Event

Definition of a single Event object is as follows.

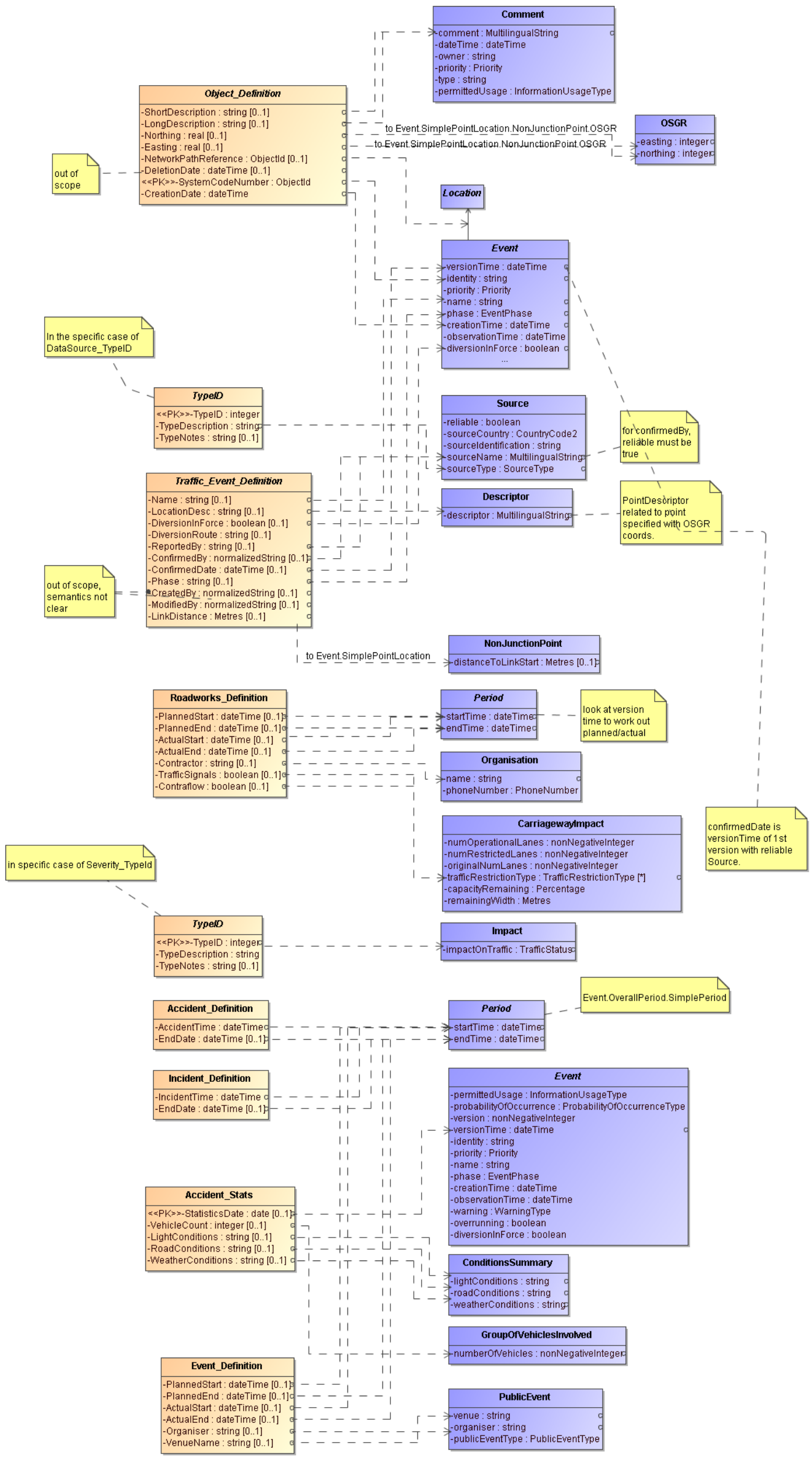
An event having a duration of at least fifteen minutes that could have a material affect on the operation of the Network. The location of an Event need not be on the Project Network in order to affect it.

Events are classified as being planned or unplanned where:

- Planned Events are those where there has been reasonable accessible prior knowledge by TCC of an Event (e.g. Roadwork, AIL)
- Unplanned Events are those that cannot be planned (e.g. Accident)

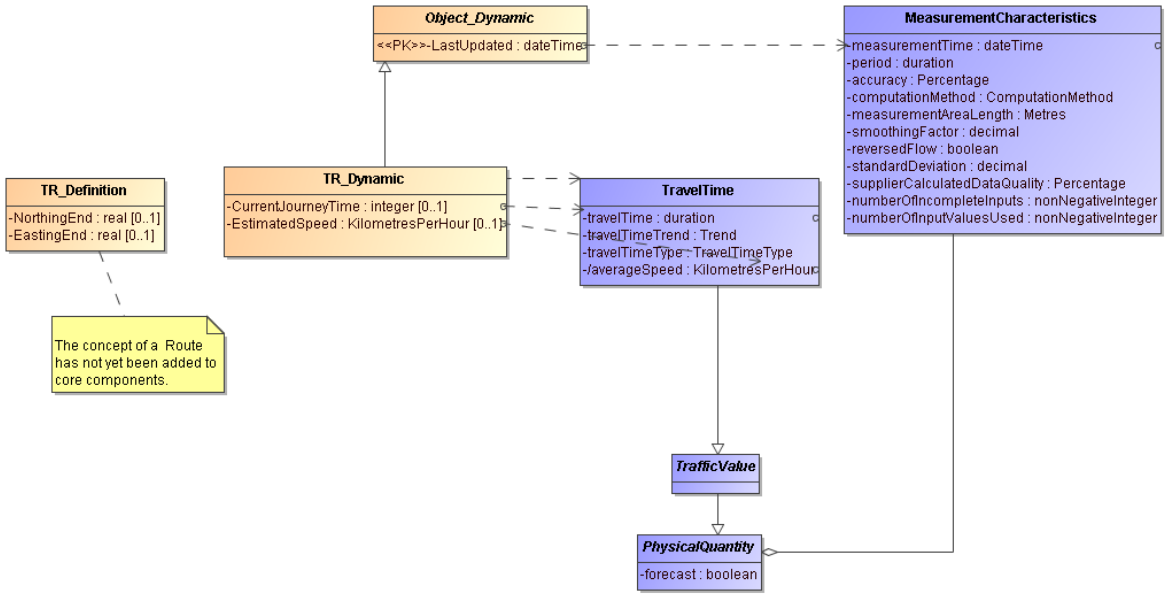


Attribute


Comment
comment : MultilingualString
-dateTime : dateTime
-owner : string
-priority : Priority
-type : string
-permittedUsage : InformationUsageType
Object_Definition
-ShortDescription : string [0..1]
-LongDescription : string [0..1]
-Northing : real [0..1]
-Easting : real [0..1]
-NetworkPathReference : ObjectId [0..1]
-DeletionDate : dateTime [0..1]
<<PK>>-SystemCodeNumber : ObjectId
-CreationDate : dateTime
to Event.SimplePointLocation.NonJunctionPoint.OSGR
to Event.SimplePointLocation.NonJunctionPoint.OSGR
OSGR
-easting : integer
-northing : integer
out of scope
Location
Event
versionTime : dateTime
identity : string
-priority : Priority
-name : string
phase : EventPhase
creationTime : dateTime
-observationTime : dateTime
diversionInForce : boolean
...
In the specific case of DataSource_TypeID
TypeID
<<PK>>-TypeID : integer
-TypeDescription : string
-TypeNotes : string [0..1]
Source
-reliable : boolean
-sourceCountry : CountryCode2
-sourceIdentification : string
sourceName : MultilingualString
sourceType : SourceType
for confirmedBy, reliable must be true
Traffic_Event_Definition
-Name : string [0..1]
-LocationDesc : string [0..1]
-DiversionInForce : boolean [0..1]
-DiversionRoute : string [0..1]
-ReportedBy : string [0..1]
-ConfirmedBy : normalizedString [0..1]
-ConfirmedDate : dateTime [0..1]
-Phase : string [0..1]
-CreatedBy : normalizedString [0..1]
-ModifiedBy : normalizedString [0..1]
-LinkDistance : Metres [0..1]
Descriptor
-descriptor : MultilingualString
PointDescriptor related to point specified with OSGR coords.
out of scope, semantics not clear
to Event.SimplePointLocation
NonJunctionPoint
distanceToLinkStart : Metres [0..1]
Roadworks_Definition
-PlannedStart : dateTime [0..1]
-PlannedEnd : dateTime [0..1]
-ActualStart : dateTime [0..1]
-ActualEnd : dateTime [0..1]
-Contractor : string [0..1]
-TrafficSignals : boolean [0..1]
-Contraflow : boolean [0..1]
Period
startTime : dateTime
endTime : dateTime
look at version time to work out planned/actual
Organisation
name : string
-phoneNumber : PhoneNumber
CarriagewayImpact
-numOperationalLanes : nonNegativeInteger
-numRestrictedLanes : nonNegativeInteger
-originalNumLanes : nonNegativeInteger
trafficRestrictionType : TrafficRestrictionType [*]
-capacityRemaining : Percentage
-remainingWidth : Metres
confirmedDate is versionTime of 1st version with reliable Source.
in specific case of Severity_TypeId
TypeID
<<PK>>-TypeID : integer
-TypeDescription : string
-TypeNotes : string [0..1]
Impact
-impactOnTraffic : TrafficStatus
Accident_Definition
-AccidentTime : dateTime
-EndDate : dateTime [0..1]
Period
startTime : dateTime
endTime : dateTime
Event.OverallPeriod.SimplePeriod
Incident_Definition
-IncidentTime : dateTime
-EndDate : dateTime [0..1]
Event
-permittedUsage : InformationUsageType
-probabilityOfOccurrence : ProbabilityOfOccurrenceType
-version : nonNegativeInteger
versionTime : dateTime
-identity : string
-priority : Priority
-name : string
-phase : EventPhase
-creationTime : dateTime
-observationTime : dateTime
-warning : WarningType
-overrunning : boolean
-diversionInForce : boolean
Accident_Stats
<<PK>>-StatisticsDate : date [0..1]
-VehicleCount : integer [0..1]
-LightConditions : string [0..1]
-RoadConditions : string [0..1]
-WeatherConditions : string [0..1]
ConditionsSummary
-lightConditions : string
-roadConditions : string
-weatherConditions : string
GroupOfVehiclesInvolved
numberOfVehicles : nonNegativeInteger
Event_Definition
-PlannedStart : dateTime [0..1]
-PlannedEnd : dateTime [0..1]
-ActualStart : dateTime [0..1]
-ActualEnd : dateTime [0..1]
-Organiser : string [0..1]
-VenueName : string [0..1]
PublicEvent
-venue : string
-organiser : string
-publicEventType : PublicEventType

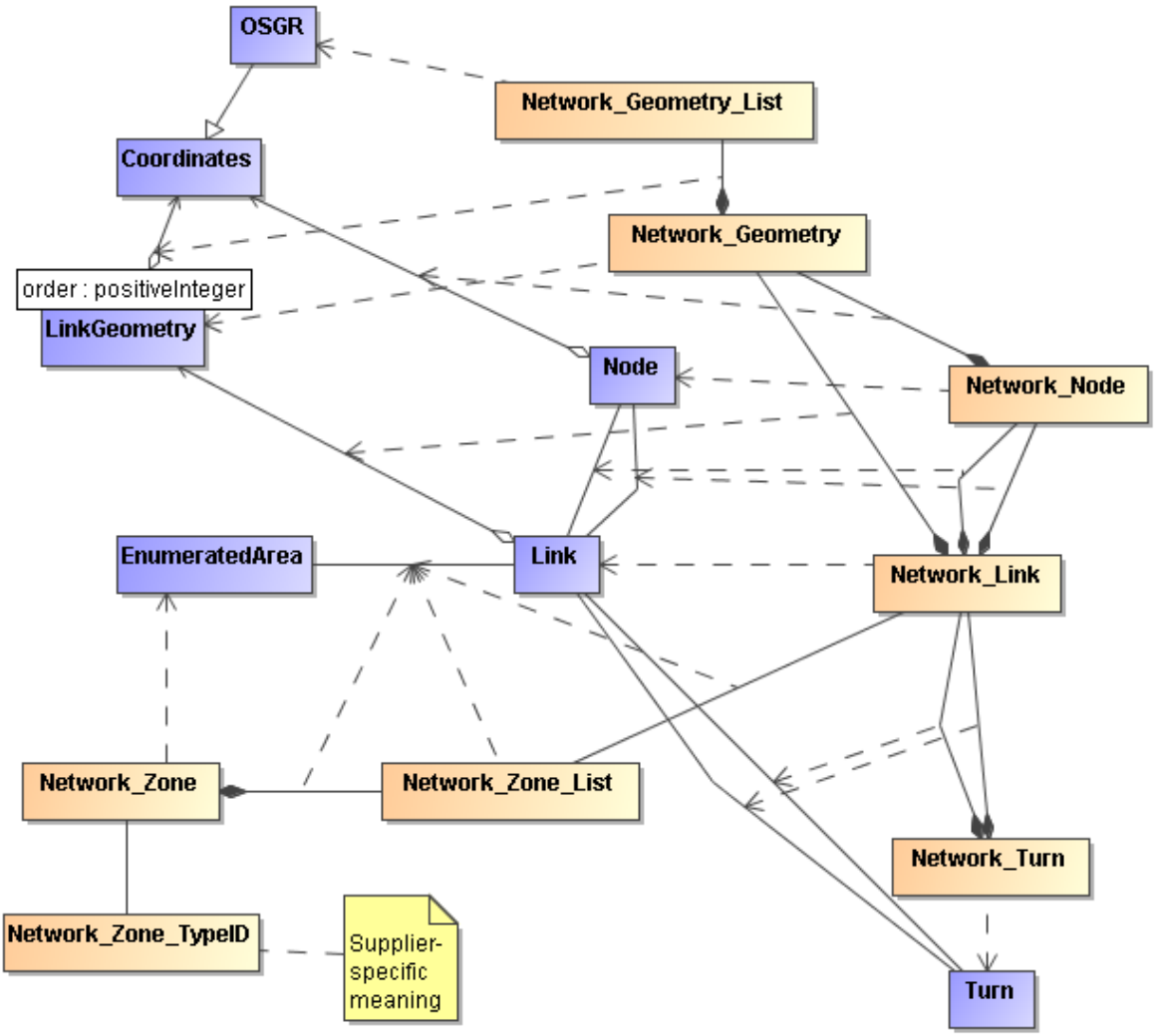
3.2.3 Transport Route Journey Time

Entity and Attribute

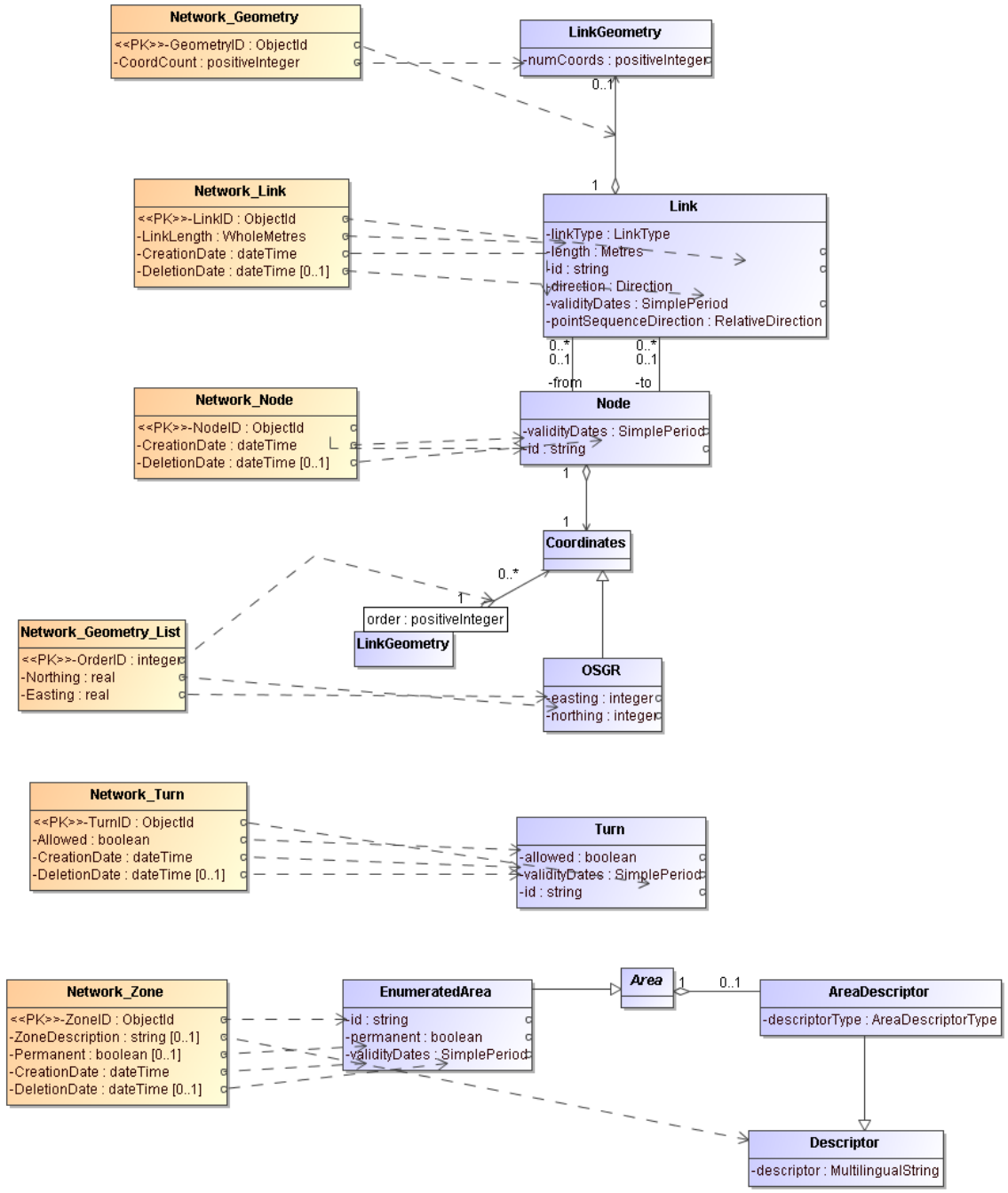


3.2.4 Location

Entity

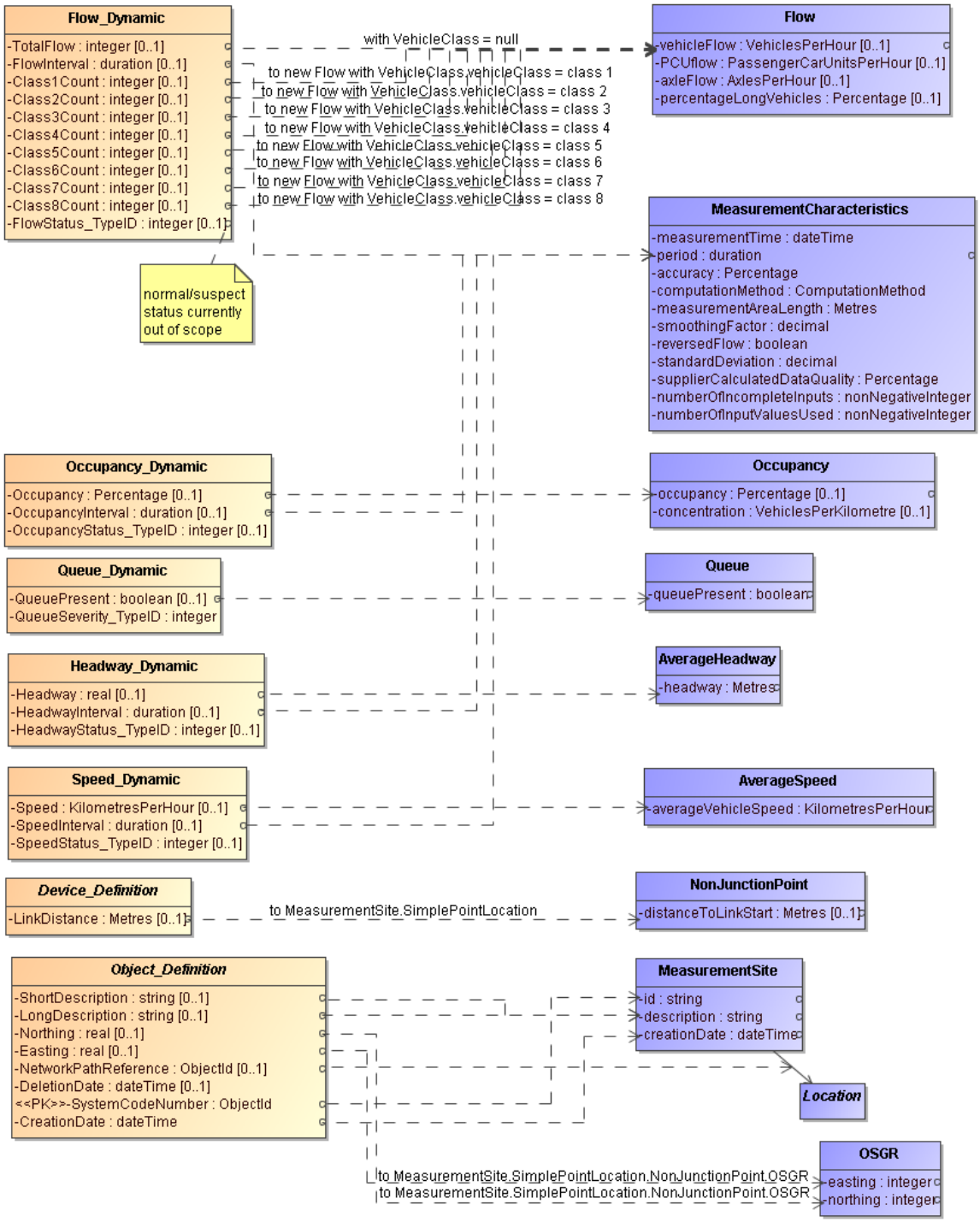


Attribute


Network_Geometry
<<PK>>-GeometryID : ObjectId
-CoordCount : positiveInteger
LinkGeometry
-numCoords : positiveInteger
0..1
1
Network_Link
<<PK>>-LinkID : ObjectId
-LinkLength : WholeMetres
-CreationDate : dateTime
-DeletionDate : dateTime [0..1]
Link
-linkType : LinkType
-length : Metres
-id : string
-direction : Direction
-validityDates : SimplePeriod
-pointSequenceDirection : RelativeDirection
0..*
0..1
-from
0..*
0..1
-to
Network_Node
<<PK>>-NodeID : ObjectId
-CreationDate : dateTime
-DeletionDate : dateTime [0..1]
Node
-validityDates : SimplePeriod
-id : string
1
1
Coordinates
0..*
1
order : positiveInteger
LinkGeometry
Network_Geometry_List
<<PK>>-OrderID : integer
-Northing : real
-Easting : real
OSGR
-easting : integer
-northing : integer
Network_Turn
<<PK>>-TurnID : ObjectId
-Allowed : boolean
-CreationDate : dateTime
-DeletionDate : dateTime [0..1]
Turn
-allowed : boolean
-validityDates : SimplePeriod
-id : string
Network_Zone
<<PK>>-ZoneID : ObjectId
-ZoneDescription : string [0..1]
-Permanent : boolean [0..1]
-CreationDate : dateTime
-DeletionDate : dateTime [0..1]
EnumeratedArea
-id : string
-permanent : boolean
-validityDates : SimplePeriod
Area
1
0..1
AreaDescriptor
-descriptorType : AreaDescriptorType
Descriptor
-descriptor : MultilingualString

3.2.5 Traffic Measurements

Entity and Attribute



3.2.6 Vehicles

UTMC concentrates on standards and protocols for interfaces between roadside units and the control centre, applications within the control centre, between control centres, and (in the future) between roadside units and vehicles.

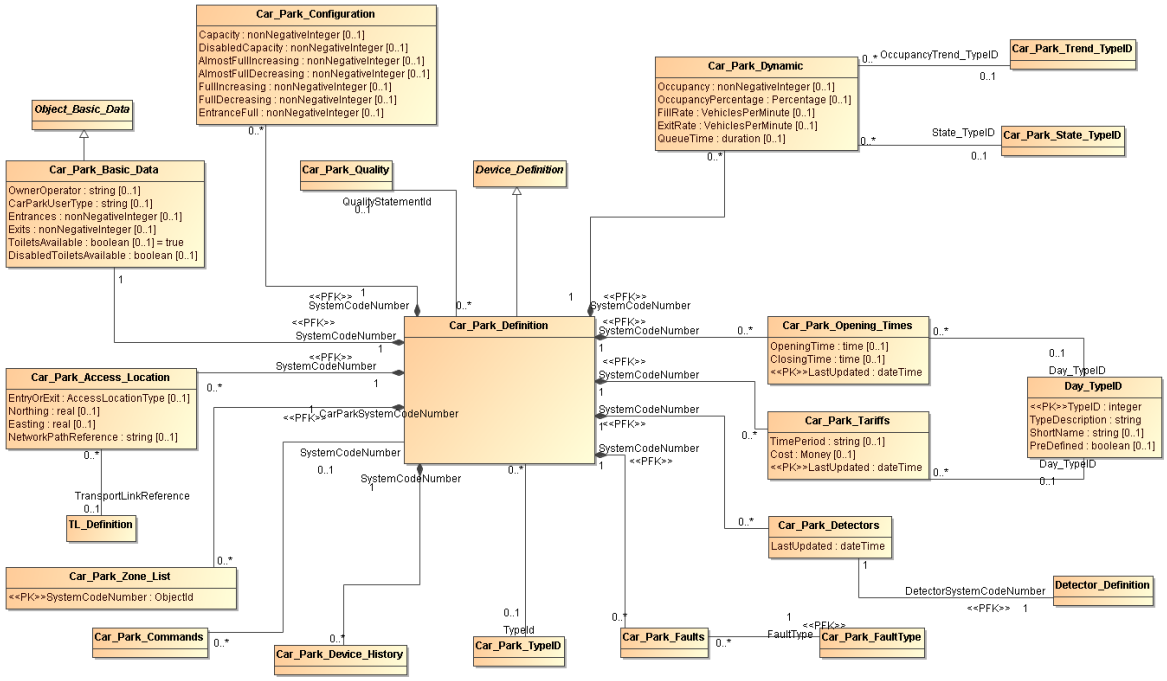
At the present time, there is no UTMC concept of vehicles and hence no mapping available.

3.2.7 Car Park Definition and Car Park Dynamic

At the present time, there is no implemented mapping between Datex II and UTMC for Car Park definition and dynamic. However, the Car Park Definition entity is shown below in terms of UTMC.

Extensions to the Datex II model to support car parks are available in Datex II although we believe (at the time of writing this document) no mapping has been done to UTMC. Parking Extensions are available at:

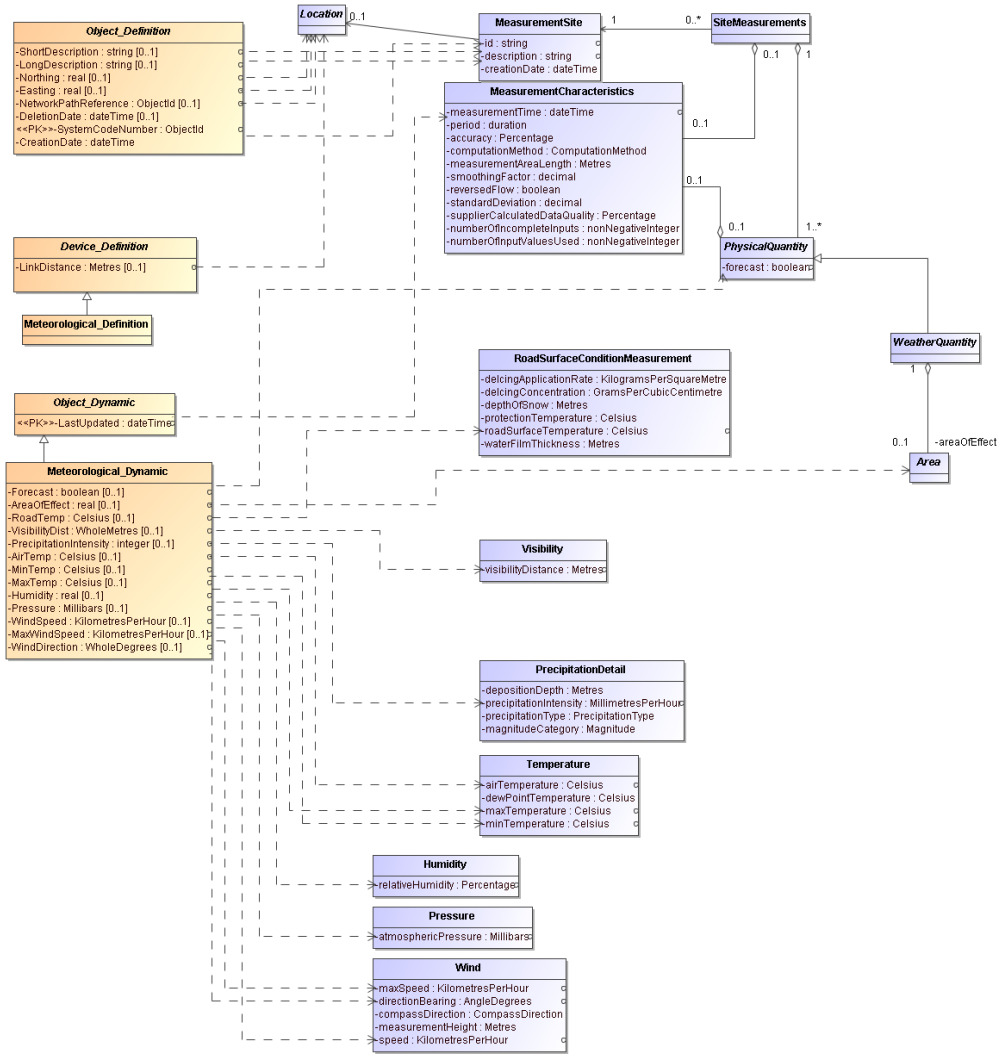
http://www.datex2.eu/sites/www.datex2.eu/files/Parking_Extension_Documentation.pdf



3.2.8 Weather

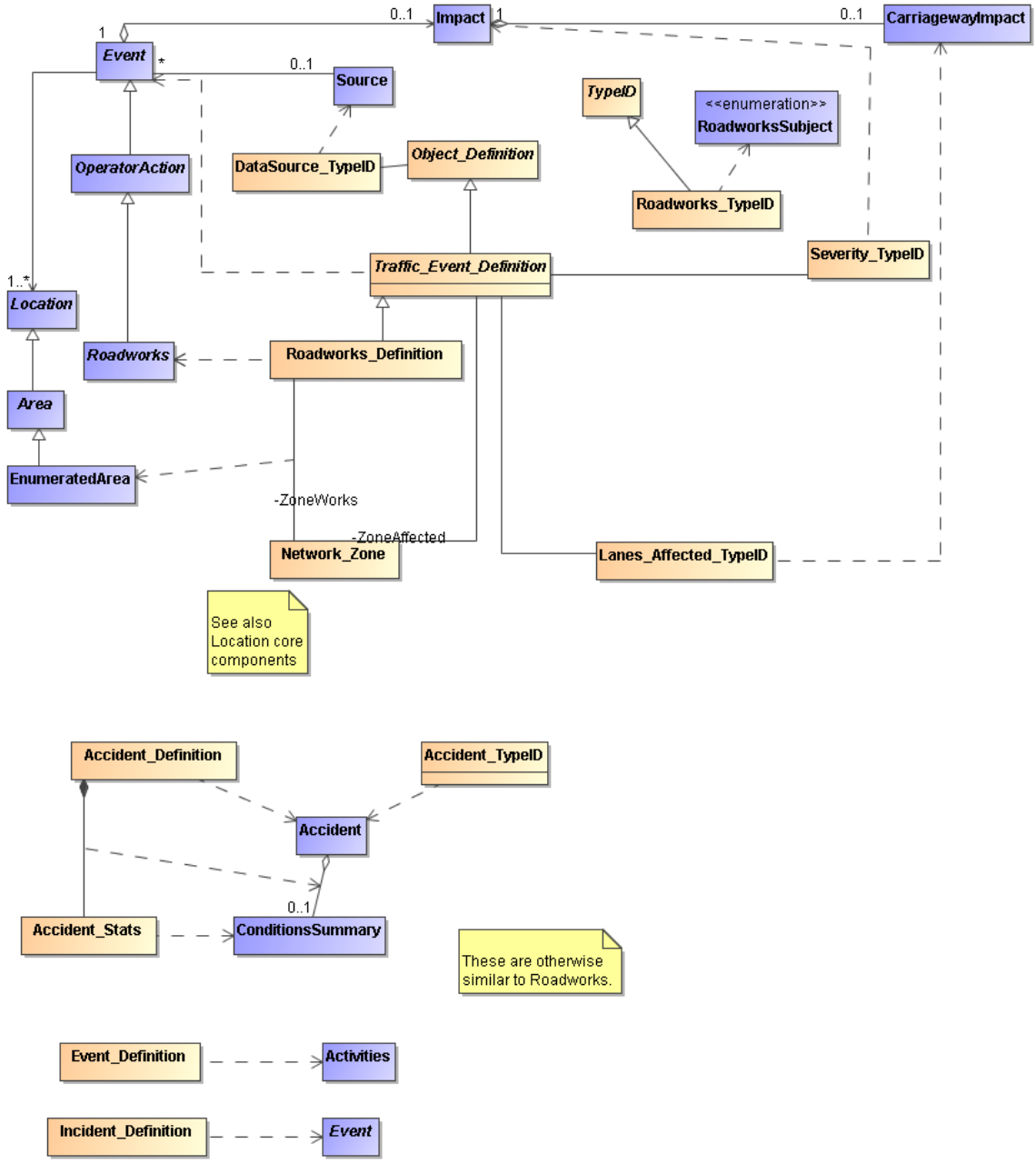
A weather element describes meteorological situations that have not been described in other sections and that may influence the road user's ability to make a safe journey. It is not the intention to provide a weather service containing the status of the weather at any time. It contains precipitation, wind and temperature sub-elements that give information on the weather conditions.

Entity and Attributes

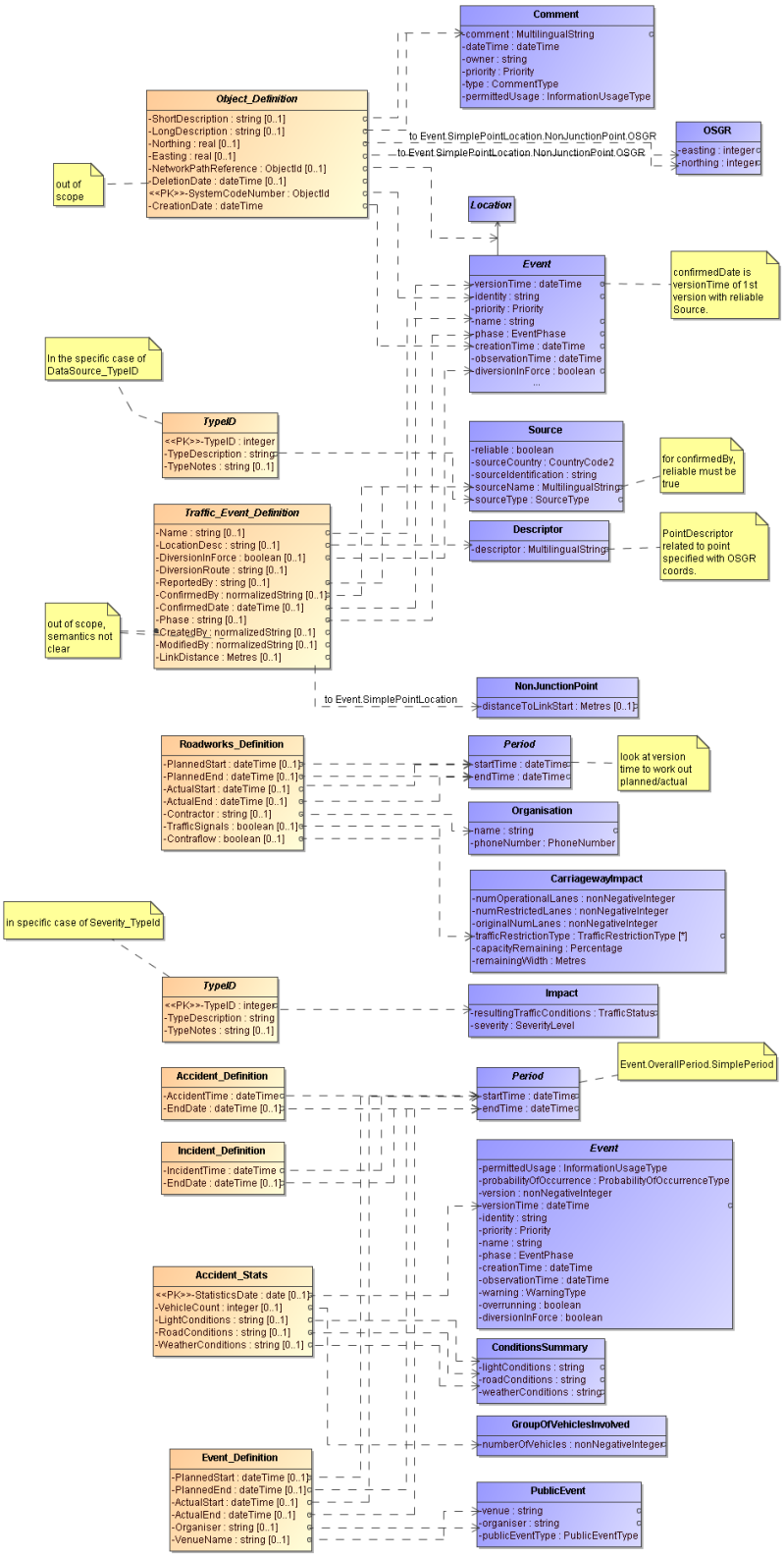


3.2.9 Roadworks/Incidents/Accident

Entity



Attribute


Comment
comment : MultilingualString
-dateTime : dateTime
-owner : string
-priority : Priority
-type : CommentType
-permittedUsage : InformationUsageType
Object_Definition
-ShortDescription : string [0..1]
-LongDescription : string [0..1]
-Northing : real [0..1]
-Easting : real [0..1]
-NetworkPathReference : ObjectId [0..1]
-DeletionDate : dateTime [0..1]
<<PK>>-SystemCodeNumber : ObjectId
-CreationDate : dateTime
to Event.SimplePointLocation.NonJunctionPoint.OSGR
to Event.SimplePointLocation.NonJunctionPoint.OSGR
OSGR
-easting : integer
-northing : integer
out of scope
Location
Event
versionTime : dateTime
identity : string
-priority : Priority
-name : string
-phase : EventPhase
-creationTime : dateTime
-observationTime : dateTime
-diversionInForce : boolean
...
confirmedDate is versionTime of 1st version with reliable Source.
In the specific case of DataSource_TypeID
TypeID
<<PK>>-TypeID : integer
-TypeDescription : string
-TypeNotes : string [0..1]
Source
-reliable : boolean
-sourceCountry : CountryCode2
-sourceIdentification : string
-sourceName : MultilingualString
-sourceType : SourceType
for confirmedBy, reliable must be true
Traffic_Event_Definition
-Name : string [0..1]
-LocationDesc : string [0..1]
-DiversionInForce : boolean [0..1]
-DiversionRoute : string [0..1]
-ReportedBy : string [0..1]
-ConfirmedBy : normalizedString [0..1]
-ConfirmedDate : dateTime [0..1]
-Phase : string [0..1]
-CreatedBy : normalizedString [0..1]
-ModifiedBy : normalizedString [0..1]
-LinkDistance : Metres [0..1]
Descriptor
descriptor : MultilingualString
PointDescriptor related to point specified with OSGR coords.
out of scope, semantics not clear
to Event.SimplePointLocation
NonJunctionPoint
-distanceToLinkStart : Metres [0..1]
Roadworks_Definition
-PlannedStart : dateTime [0..1]
-PlannedEnd : dateTime [0..1]
-ActualStart : dateTime [0..1]
-ActualEnd : dateTime [0..1]
-Contractor : string [0..1]
-TrafficSignals : boolean [0..1]
-Contraflow : boolean [0..1]
Period
startTime : dateTime
endTime : dateTime
look at version time to work out planned/actual
Organisation
-name : string
-phoneNumber : PhoneNumber
CarriagewayImpact
-numOperationalLanes : nonNegativeInteger
-numRestrictedLanes : nonNegativeInteger
-originalNumLanes : nonNegativeInteger
-trafficRestrictionType : TrafficRestrictionType [*]
-capacityRemaining : Percentage
-remainingWidth : Metres
in specific case of Severity_TypeId
TypeID
<<PK>>-TypeID : integer
-TypeDescription : string
-TypeNotes : string [0..1]
Impact
-resultingTrafficConditions : TrafficStatus
-severity : SeverityLevel
Accident_Definition
-AccidentTime : dateTime
-EndDate : dateTime [0..1]
Period
startTime : dateTime
endTime : dateTime
Event.OverallPeriod.SimplePeriod
Incident_Definition
-IncidentTime : dateTime
-EndDate : dateTime [0..1]
Event
-permittedUsage : InformationUsageType
-probabilityOfOccurrence : ProbabilityOfOccurrenceType
-version : nonNegativeInteger
-versionTime : dateTime
-identity : string
-priority : Priority
-name : string
-phase : EventPhase
-creationTime : dateTime
-observationTime : dateTime
-warning : WarningType
-overrunning : boolean
-diversionInForce : boolean
Accident_Stats
<<PK>>-StatisticsDate : date [0..1]
-VehicleCount : integer [0..1]
-LightConditions : string [0..1]
-RoadConditions : string [0..1]
-WeatherConditions : string [0..1]
ConditionsSummary
-lightConditions : string
-roadConditions : string
-weatherConditions : string
GroupOfVehiclesInvolved
-numberOfVehicles : nonNegativeInteger
Event_Definition
-PlannedStart : dateTime [0..1]
-PlannedEnd : dateTime [0..1]
-ActualStart : dateTime [0..1]
-ActualEnd : dateTime [0..1]
-Organiser : string [0..1]
-VenueName : string [0..1]
PublicEvent
-venue : string
-organiser : string
-publicEventType : PublicEventType

3.2.10 Optional Transport Links

Whilst there is no direct mappings available between UTMC and Datex II to Transport Links, Cloud Amber have mapped the following tables from UTMC to Datex II which are optionally available:

- TL_Definition
- TL_ANPR_Dynamic

4 Developers Reference – Datex II

4.1 Introduction

The purpose of this section is to provide a reference for software developers to provide the necessary XML to interface to the Datex II data feed and make use of its functionality.

4.2 Accessing the Datex II data feed

The Datex II data feed is accessible through the following interfaces:

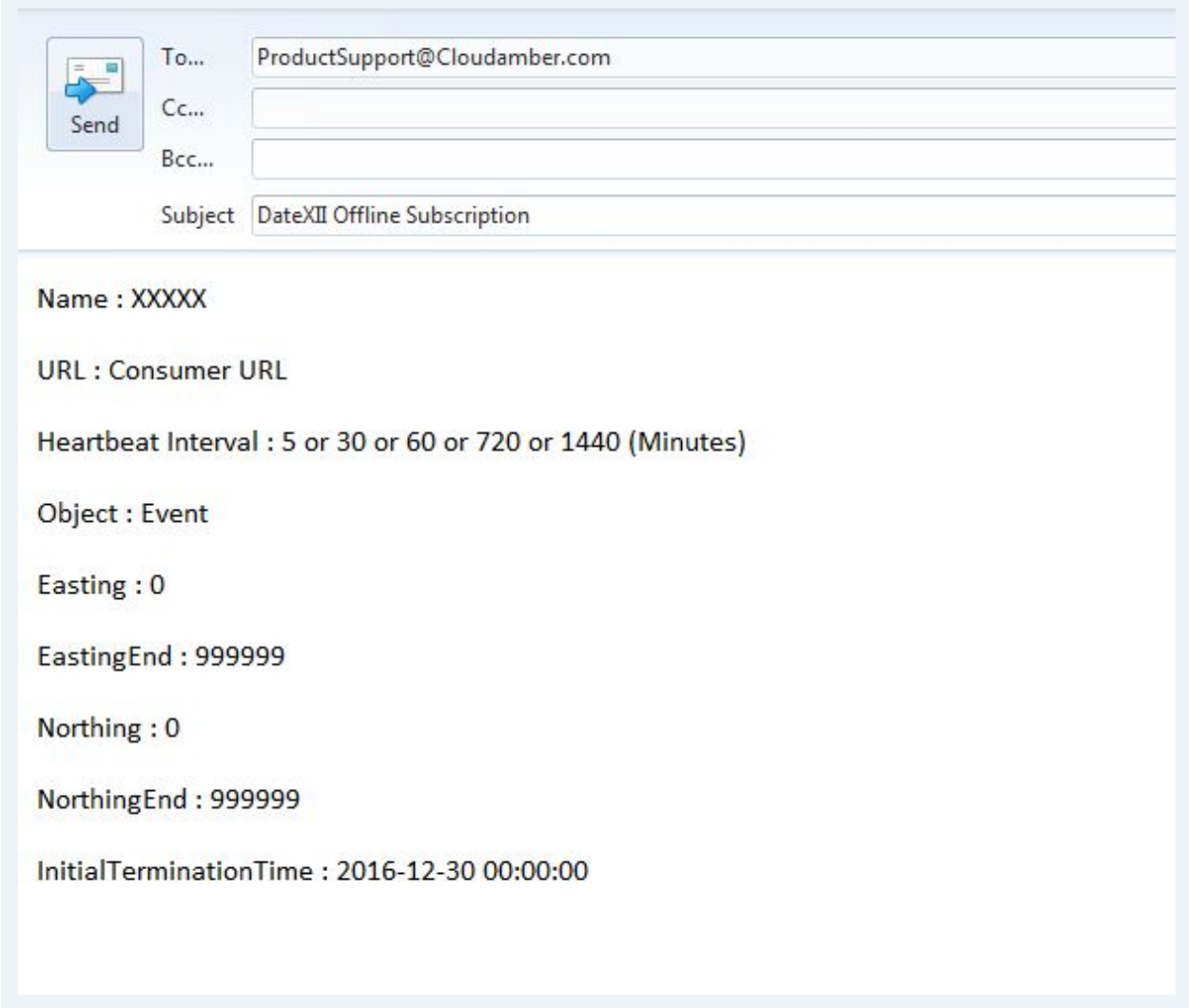
- WCF (Windows Communication Foundation using .Net)
- Web Service (SOAP – Simple Object Access Protocol)
- HTTPGet

DateXII feed can also support the offline subscription mechanism.

Customer need to send the following details to the **productsupport@cloudamber.com**.

The product support team will register this request and then initiate the data sharing until the termination time.

- Name
- Consumer URL
- Heartbeat Interval
- Object
- Easting
- EastingEnd
- Northing
- NorthingEnd
- InitialTerminationTime



Heartbeat Notification

Using a Heartbeat message, the Notification Producer sends a regular notification message to the Consumer based on heartbeat interval until the subscription is in active state to show that it is still active, even if no update messages have been sent. This removes the need for the Consumer to poll, as the Consumer can detect a potential failure by the failure of a Heartbeat to arrive within the subscribed heartbeat interval.

```
<?xml version="1.0"?>
- <HeartbeatNotification>
    <RequestTimestamp>2015-12-09T14:38:48+00:00</RequestTimestamp>
    <ProducerRef>Cloud Test</ProducerRef>
    <Status>true</Status>
    <ServiceStartedTime>2015-12-09T14:38:48+00:00</ServiceStartedTime>
</HeartbeatNotification>
```

The Databroker publish the following list of objects in Datex II format:

- Events
- Road Works
- Traffic Data
- VMS
- Car Park
- CCTV
- Matrix Signal
- Journey Times – Current(Transport Route)
- Journey Times – Predictive
- PredefinedLocation – Traffic Data Links
- PredefinedLocation – Traffic Data Section
- PredefinedLocation – VMS
- PredefinedLocation – Matrix
- PredefinedLocation – Journey Time Sections
- PredefinedLocation – CCTV

By extending the Datex II format, Databroker can also publish the following objects,

- Traffic Data Extension
    - Queue Dynamic
    - Scoot Dynamic
- Cycle Hubs
- EV Charging Points
- Walkers Facilities
- Point Of Interest
- Meteorological

The above mentioned objects are accessible through different methods as follows:

Object Name	Method Name	Results
Events	GetDateXIIEvents	Returns list of Events which meets filter [Table 5.2] criteria from 1 to 6.
Road Works	GetDateXIIRoadWorks	Returns list of Road works which meets filter [Table 5.2] criteria from 1 to 6.
Traffic Data	GetDateXIITrafficData	Returns list of Traffic Data which meets filter [Table 5.2] criteria from 1 to 4.
Traffic Data Extension	GetDateXIITrafficDataExtension	Returns list of Queue & Scoot data which meets filter [Table 5.2] criteria from 1 to 4.
Variable Message Sign	GetDateXIIAllVMS	Returns list of VMS which meets filter [Table 5.2] criteria from 1 to 4.
Car Park(DatexII version 1.0)	GetDatexIICarPark	Returns list of Car Park data which meets filter [Table 5.2] criteria from 1 to 4.
Car Park(DatexII version 2.2.0)	GetDatexIICarPark20	Returns list of Car Park data meets filter [Table 5.2] criteria from 1 to 4.
CCTV	GetDatexIICCTV	Returns list of CCTV which meets filter [Table 5.2] criteria from 1 to 4.

Object Name	Method Name	Results
Journey Times-Current	GetDatexIITransportRouteJourneyTime	Returns list of Current Journey Times which meets filter [Table 5.2] criteria from 1 to 4.
Journey Times-Predictive	GetDatexIITransportRoutePredictiveJourneyTimes	Returns list of Journey Times - Predictive which meets filter [Table 5.2] criteria 7.

Cycle Hubs	GetDatexIICycleHubs	Returns list of Cycle Hubs which meets filter [Table 5.2] criteria from 1 to 4.
EV Charging Points	GetDatexIIEVChargingPoints	Returns list of EV Charging Points which meets filter [Table 5.2] criteria from 1 to 4.
Walkers Facilities	GetDatexIIWalkersFacilities	Returns list of Walkers Facilities which meets filter [Table 5.2] criteria from 1 to 4.
Point Of Interest	GetDatexIIPointOfInterest	Returns list of Point Of Interest which meets filter [Table 5.2] criteria from 1 to 4.
Meteorological	GetDatexIIMeteoroLogical	Returns list of Meteorological which meets filter [Table 5.2] criteria from 1 to 4.
PredefinedLocation-Traffic Data Links	GetDateXIIPredefinedLocation	Returns list of Traffic Data Links location which meets filter [Table 5.2] criteria 8. E.g:type=link
PredefinedLocation-Traffic Data Section	GetDateXIIPredefinedLocation	Returns list of Traffic Data section location which meets filter [Table 5.2] criteria 8. E.g:type=section
PredefinedLocation-VMS	GetDateXIIPredefinedLocation	Returns list of VMS location which meets filter [Table 5.2] criteria 8. E.g:type=vms
PredefinedLocation-Matrix	GetDateXIIPredefinedLocation	Returns list of Matrix Signal location which meets filter [Table 5.2] criteria 8. E.g:type=matrix
PredefinedLocation-Journey Time Sections	GetDateXIIPredefinedLocation	Returns list of Journey Time Section location which meets filter [Table 5.2] criteria 8. E.g:type=tr
PredefinedLocation-CCTV	GetDateXIIPredefinedLocation	Returns list of CCTV location which meets filter [Table 5.2] criteria 8. E.g:type=cctv

Table 5.1

Each of the feeds requires a default set of input parameters which are shown in the table below.

S.No	Parameter Name	Definition	Example
1	Easting	Boundary Point	305750
2	EastingEnd	Boundary Point	429125
3	Northing	Boundary Point	0
4	NorthingEnd	Boundary Point	144500
5	StartDate	Start date of the required object activity	28/09/2011
6	EndDate	End date of the required object activity	28/09/2011
7	Value	Object Predictive Hours value	1
8	Type	Type of object activity	link(or)section(or)vms(or)matrix(or)tr(or)cctv

Table 5.2

Note that you will receive data from the DatexII feed where the following conditions are met:

1.@StartDate <=IncidentTime and @EndDate >=EndDate

2.@StartDate <= IncidentTime and @EndDate <=EndDate

3.@StartDate >= IncidentTime and @EndDate <=EndDate

4.@EndDate >=IncidentTime and EndDate is NULL

Where @StartDate @EndDate are your input parameters.

The version of Datex II supported by Databroker for each object is shown below:

Object Name	DateXII Version
Events	1.0
Road Works	1.0
Traffic Data	1.0
Traffic Data Extension	1.0 Extension
Variable Message Sign	1.0
Car Park(DatexII version 1.0)	1.0
Car Park(DatexII version 2.2.0)	2.2.0
CCTV	1.0
Matrix Signal	1.0
Journey Times-Current	1.0
Journey Times-Predictive	1.0
Cycle Hubs	1.0 Extension
EV Charging Points	1.0 Extension
Walkers Facilities	1.0 Extension
Point Of Interest	1.0 Extension
Meteorological	1.0 Extension
PredefinedLocation-Traffic Data Links	1.0

PredefinedLocation-Traffic Data Section	1.0
PredefinedLocation-VMS	1.0
PredefinedLocation-Matrix	1.0
PredefinedLocation-Journey Time Sections	1.0
PredefinedLocation-CCTV	1.0

Table 5.3

XML Schema:
The XSD file can be found in the following location the relevant versions of Datex II:

Version 1.0

http://labs.cloudamberdevelopment.com/Datexii1.0/XsdFiles/xsd.zip

Version 2.2.0

http://labs.cloudamberdevelopment.com/Datexii2.0/XsdFiles/xsd.zip

Version 1.0 Extension

http://labs.cloudamberdevelopment.com/DatexiiExtension/XsdFiles/xsd.zip

Class Files:

C# class file can be found in the following location:

Version 1.0

http://labs.cloudamberdevelopment.com/Datexii1.0/ClassFiles/Class.zip

Version 2.2.0

http://labs.cloudamberdevelopment.com/Datexii2.0/ClassFiles/Class.zip

Version 1.0 Extension

http://labs.cloudamberdevelopment.com/DatexiiExtension/ClassFiles/Class.zip

Section 4.4 explains about usage of the classes listed above.

This parameter is applicable for all the interfaces except FTP.

The FTP folder will have DatexII format xml files for all available data without any filters being applied to them.

Additionally; Data Broker has one more common method for all the above mentioned objects, the only limitation being that the return type is a string in xml format. The method name is "GetDateXIIData" and it requires one more additional parameter about required object name.

4.3 URL Format

Each interfaces can be accessible through different url, the details of which are as follows,

- **WCF**
  http://[Ipaddress or Hostname]:port/CloudAmber/DateXII/wcf
- **Web Service (SOAP – Simple Object Access Protocol)** -
  http://[Ipaddress or Hostname]:port/CloudAmber/DateXII/Soap
- **HTTP Get**
  POX : http://[Ipaddress or Hostname]:port/CloudAmber/DateXII/pox/MethodName?easting=[easting value]&eastingEnd=[easting end value]&Northing=[northing value]&NorthingEnd=[northing end value]&startDate=[start date value]&endDate=[end date value]&value=[hours value]&type=[type value]

  JSON : http://[Ipaddress or Hostname]:port/CloudAmber/DateXII/Json/MethodName?easting=[easting value]&eastingEnd=[easting end value]&Northing=[northing value]&NorthingEnd=[northing end value]&startDate=[start date value]&endDate=[end date value]& value=[hours value]&type=[type value]

- **FTP**
  Ftp ipaddress or host name

4.4 Input and Output for each interface

So as to access the data, we have provided a series of examples below for all available interfaces mentioned in the previous section;

To provide some uniformity, each example is broken down as follows:

- URL to subscribe to (the ip address and port will be different for each customer)
- Input XML
- Output XML

Processing Output:

The output XML from "GetDateXIIData" method can be converted into class object by using following c# code:

```
byte[] byteArray = Encoding.ASCII.GetBytes("Return XML from DateXII");

MemoryStream stream = new MemoryStream(byteArray);

XmlSerializer serializer = new XmlSerializer(typeof(Namespace.D2LogicalModel));

Namespace.D2LogicalModel data= new Namespace.D2LogicalModel();

data= (Namespace.D2LogicalModel)serializer.Deserialize(stream);
```

4.4.1 WCF Request and Response

The proxy class can be created by using following command,

"svcutil http://[Ipaddress or Hostname]:port/CloudAmber/DateXII"

By using the generated proxy class, the required methods can be called as follows,

URL	http://ipaddress:port/CloudAmber/DateXII/wcf
Input XML	**DateXIIClient dateXIIClient = new DateXIIClient();** string datexiiXML = dateXIIClient.**GetDateXIIData**("Roadworks",305750,429125,0,144500,Convert.ToDateTime("28/09/2010"),Convert.ToDateTime("28/09/2011"),''); The other methods (**Table 5.1**) can be called as same like above code.
Output XML	<?xml version="1.0" encoding="utf-8" ?> <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance/" xmlns:xsd="http://www.w3.org/2001/XMLSchema/" ><soapenv:Body> <d2LogicalModel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" modelBaseVersion="1.0" xmlns="http://datex2.eu/schema/1_0/1_0"> <exchange> <supplierIdentification> <country>gb</country> <nationalIdentifier>NTCC</nationalIdentifier>

	</supplierIdentification> </exchange> <payloadPublication xsi:type="SituationPublication" lang="en"> <publicationTime>2011-09-28T12:35:58+01:00</publicationTime> </payloadPublication> </d2LogicalModel></soapenv:Body></soapenv:Envelope>

4.4.2 Web Service: SOAP Request and Response

URL	http://ipaddress:port/CloudAmber/DateXII/soap
Input XML	<?xml version="1.0" encoding="utf-8"?> <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"> <SOAP-ENV:Body> <GetDateXIIData xmlns="http://www.CloudAmber.com"><dateXIIType>UnPlannedEvents</dateXIIType><easting>30</easting><eastingEnd>543543</eastingEnd><northing>0</northing><northingEnd>543534</northingEnd><startDate>2011-09-28T00:00:00.0000000+01:00</startDate><endDate>2011-09-28T00:00:00.0000000+01:00</endDate></GetDateXIIData></SOAP-ENV:Body></SOAP-ENV:Envelope> The other methods (Table 5.1) can be called as same like above xml.
Output XML	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"> <s:Body> <GetDateXIIDataResponse xmlns="http://www.CloudAmber.com"><GetDateXIIDataResult><?xml version="1.0" encoding="utf-8" ?> <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance/" xmlns:xsd="http://www.w3.org/2001/XMLSchema/" ><soapenv:Body>
 <d2LogicalModel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" modelBaseVersion="1.0" xmlns="http://datex2.eu/schema/1_0/1_0">
 <exchange>

	<supplierIdentification>
 <country>gb</country>
 <nationalIdentifier>NTCC</nationalIdentifier>
 </supplierIdentification>
 </exchange>
 <payloadPublication xsi:type="SituationPublication" lang="en">
 <publicationTime>2011-09-28T12:35:59+01:00</publicationTime>
 </payloadPublication>
 </d2LogicalModel></soapenv:Body></soapenv:Envelope></GetDateXIIDataResult></GetDateXIIDataResponse></s:Body> </s:Envelope>

4.4.3 HTTP GET- REST- POX

URL for POX	http://[Ipaddress or Hostname]:port/CloudAmber/DateXII/pox/**MethodName**?easting=[easting value]&eastingEnd=[easting end value]&Northing=[northing value]&NorthingEnd=[northing end value]&startDate=[start date value]&endDate=[end date value] The other methods (**Table 5.1**) can be called as same like above xml.
Output XML	<?xml version="1.0" encoding="utf-8"?><RoadWorksd2LogicalModel modelBaseVersion="1.0" schemaLocation="http://datex2.eu/schema/1_0/1_0 http://datex2.eu/schema/1_0/1_0/DATEXIISchema_1_0_1_0.xsd" xmlns="http://datex2.eu/schema/1_0/1_0/RoadWorks" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"><exchange xmlns="http://datex2.eu/schema/1_0/1_0"><supplierIdentification><country>gb</country><nationalIdentifier>NTCC</nationalIdentifier></supplierIdentification></exchange><payloadPublication type="SituationPublication" lang="en" xmlns="http://datex2.eu/schema/1_0/1_0"><publicationTime>2011-09-29T15:32:59+01:00</publicationTime><publicationCreator><country>gb</country><nationalIdentifier>NTCC</nationalIdentifier></publicationCreator><situation id="GUID0"><headerInformation><areaOfInterest>national</areaOfInterest><confidentiality>restrictedToAuthoritiesTrafficOperatorsAndPublishers</confidentiality><informationUsage>broadcast</informationUsage><informationStatus>real</informationStatus></headerInformation><situationRecord type="MaintenanceWorks"

	id="GUID489843264"><situationRecordCreationReference>P-09-1224-0190</situationRecordCreationReference><situationRecordCreationTime>2011-08-17T13:02:29</situationRecordCreationTime><situationRecordVersion>4</situationRecordVersion><situationRecordVersionTime>0001-01-01T00:00:00</situationRecordVersionTime><situationRecordFirstSupplierVersionTime>0001-01-01T00:00:00</situationRecordFirstSupplierVersionTime><probabilityOfOccurrence>certain</probabilityOfOccurrence><sourceInformation><sourceCountry>gb</sourceCountry><sourceName><value lang="en">BBC</value></sourceName></sourceInformation><validity><validityStatus>definedByValidityTimeSpec</validityStatus><validityTimeSpecification><overallStartTime>2011-08-19T20:00:00</overallStartTime><overallEndTime>2011-08-20T06:00:00</overallEndTime><validPeriod><startOfPeriod>2011-08-19T20:00:00</startOfPeriod><endOfPeriod>2011-08-20T06:00:00</endOfPeriod></validPeriod></validityTimeSpecification></validity><impact><impactOnTraffic>Severe</impactOnTraffic><impactDetails><capacityRemaining>100</capacityRemaining><numberOfLanesRestricted>0</numberOfLanesRestricted><numberOfOperationalLanes>2</numberOfOperationalLanes><originalNumberOfLanes>2</originalNumberOfLanes><trafficRestrictionType>carriagewayPartiallyObstructed</trafficRestrictionType></impactDetails><delays><delayTimeValue>600</delayTimeValue></delays></impact><nonGeneralPublicComment><comment><value lang="en">The A1 southbound exit slip to the A638 will be closed, due to barrier repairs, from 8 pm on 19 August 2011 to 6 am on 20 August 2011.</value></comment></nonGeneralPublicComment><groupOfLocations><locationContainedInGroup type="Point"><tpegpointLocation type="TPEGSimplePoint"><tpegDirection>A1 southbound exit for A1(M)/A638 southBound</tpegDirection><tpegLocationType>intersection</tpegLocationType><point type="TPEGJunction"><pointCoordinates><latitude>53.5789704361836</latitude><longitude>-1.20954269961825</longitude></pointCoordinates><name><descriptor><value lang="en">A1 southbound exit for A1(M)/A638 southBound</value></descriptor></name><ilc><descriptor><value lang="en">A1</value></descriptor></ilc></point></tpegpointLocation></locationContainedInGroup></groupOfLocations><roadMaintenanceType>Highways - BBC</roadMaintenanceType></situationRecord></situation></payloadPublication></RoadWorksd2LogicalModel>

4.4.1 HTTP GET- REST- JSON

URL for POX	http://[Ipaddress or Hostname]:port/CloudAmber/DateXII/Json/MethodName?easting=[easting value]&eastingEnd=[easting end value]&Northing=[northing value]&NorthingEnd=[northing end value]&startDate=[start date value]&endDate=[end date value] The other methods (Table 5.1) can be called as same like above xml.
Output XML	{"d":{"__type":"d2LogicalModel:#CurrentRoadWorks","exchangeField":{"__type":"d2LogicalModelExchange:#CurrentRoadWorks","supplierIdentificationField":{"__type":"d2LogicalModelExchangeSupplierIdentification:#CurrentRoadWorks","countryField":"gb","nationalIdentifierField":"NTCC"}},"modelBaseVersionField":1.0,"payloadPublicationField":{"__type":"d2LogicalModelPayloadPublication:#CurrentRoadWorks","langField":"en","publicationCreatorField":{"__type":"d2LogicalModelPayloadPublicationPublicationCreator:#CurrentRoadWorks","countryField":"gb","nationalIdentifierField":"NTCC"},"publicationTimeField":"2011-09-29T15:33:59+01:00","situationField":[{"__type":"d2LogicalModelPayloadPublicationSituation:#CurrentRoadWorks","headerInformationField":{"__type":"d2LogicalModelPayloadPublicationSituationHeaderInformation:#CurrentRoadWorks","areaOfInterestField":"national","confidentialityField":"restrictedToAuthoritiesTrafficOperatorsAndPublishers","informationStatusField":"real","informationUsageField":"broadcast"},"idField":"GUID0","situationRecordField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecord:#CurrentRoadWorks","effectOnRoadLayoutField":null,"groupOfLocationsField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocations:#CurrentRoadWorks","locationContainedInGroupField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroup:#CurrentRoadWorks","tpegpointLocationField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocation:#CurrentRoadWorks","framedPointField":null,"fromField":null,"pointField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPoint:#CurrentRoadWorks","ilcField":[{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointIlc:#CurrentRoadWorks","descriptorField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointIlcDescriptor:#CurrentRoadWorks","valueField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointIlcDescriptorValue:#CurrentRoadWorks","langField":"en","valueField":"A1"}},"tpegDescriptorTypeField":null}],"nameField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointName:#CurrentRoadWorks","descriptorField":{"__type":"d2LogicalModelPayloadPublic

ationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointNameDescriptor:#CurrentRoadWorks","valueField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointNameDescriptorValue:#CurrentRoadWorks","langField":"en","valueField":"A1 southbound exit for A1(M)\/A638 southBound"}},"tpegDescriptorTypeField":null},"pointCoordinatesField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordGroupOfLocationsLocationContainedInGroupTpegpointLocationPointPointCoordinates:#CurrentRoadWorks","latitudeField":53.5789704361836,"longitudeField":-1.20954269961825},"typeField":"TPEGJunction"},"toField":null,"tpegDirectionField":"A1 southbound exit for A1(M)\/A638 southBound","tpegLocationTypeField":"intersection","typeField":"TPEGSimplePoint"},"typeField":"Point"}},"idField":"GUID489843264","impactField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordImpact:#CurrentRoadWorks","delaysField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordImpactDelays:#CurrentRoadWorks","delayTimeValueField":600},"impactDetailsField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordImpactImpactDetails:#CurrentRoadWorks","capacityRemainingField":100,"numberOfLanesRestrictedField":0,"numberOfOperationalLanesField":2,"originalNumberOfLanesField":2,"trafficRestrictionTypeField":"carriagewayPartiallyObstructed"},"impactOnTrafficField":"Severe"},"nonGeneralPublicCommentField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordNonGeneralPublicComment:#CurrentRoadWorks","commentField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordNonGeneralPublicCommentComment:#CurrentRoadWorks","valueField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordNonGeneralPublicCommentCommentValue:#CurrentRoadWorks","langField":"en","valueField":"The A1 southbound exit slip to the A638 will be closed , due to barrier repairs, from 8 pm on 19 August 2011 to 6 am on 20 August 2011."}}},"probabilityOfOccurrenceField":"certain","roadMaintenanceTypeField":"Highways - BBC","situationRecordCreationReferenceField":"P-09-1224-0190","situationRecordCreationTimeField":"\/Date(1313582549000+0100)\/","situationRecordFirstSupplierVersionTimeField":"\/Date(-62135596800000+0000)\/","situationRecordVersionField":"4","situationRecordVersionTimeField":"\/Date(-62135596800000+0000)\/","sourceInformationField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordSourceInformation:#CurrentRoadWorks","sourceCountryField":"gb","sourceNameField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordSourceInformationSourceName:#CurrentRoadWorks","valueField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordSourceInformationSourceNameValue:#CurrentRoadWorks","langField":"en","valueField":"BBC"}}},"typeField":"MaintenanceWorks","validityField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordValidity:#CurrentRoadWorks","validityStatusField":"definedByValidityTimeSpec","validityTimeSpecificationField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordVali

	dityValidityTimeSpecification:#CurrentRoadWorks","overallEndTimeField":"\/Date(1313816400000+0100)\/","overallStartTimeField":"\/Date(1313780400000+0100)\/","validPeriodField":{"__type":"d2LogicalModelPayloadPublicationSituationSituationRecordValidityValidityTimeSpecificationValidPeriod:#CurrentRoadWorks","endOfPeriodField":"\/Date(1313816400000+0100)\/","startOfPeriodField":"\/Date(1313780400000+0100)\/"}}}}}],"typeField":"SituationPublication"},"schemaLocationField":"http:\/\/datex2.eu\/schema\/1_0\/1_0 http:\/\/datex2.eu\/schema\/1_0\/1_0\/DATEXIISchema_1_0_1_0.xsd"}}

4.4.2 FTP

The FTP folder will be populated with the following XML files and its update frequency will be every 10 minutes.

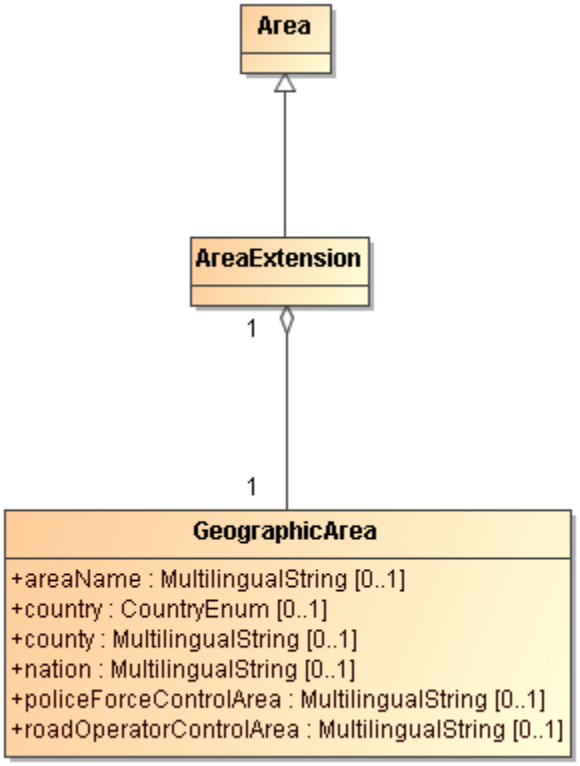
- EventsddMMyyyyHHmm.xml
- RoadWorksddMMyyyyHHmm.xml
- TrafficDataddMMyyyyHHmm.xml
- VeriableMessageSigndMMyyyyHHmm.xml
- PredefinedLocationddMMyyyyHHmm.xml

The FTP server details will be provided upon request.

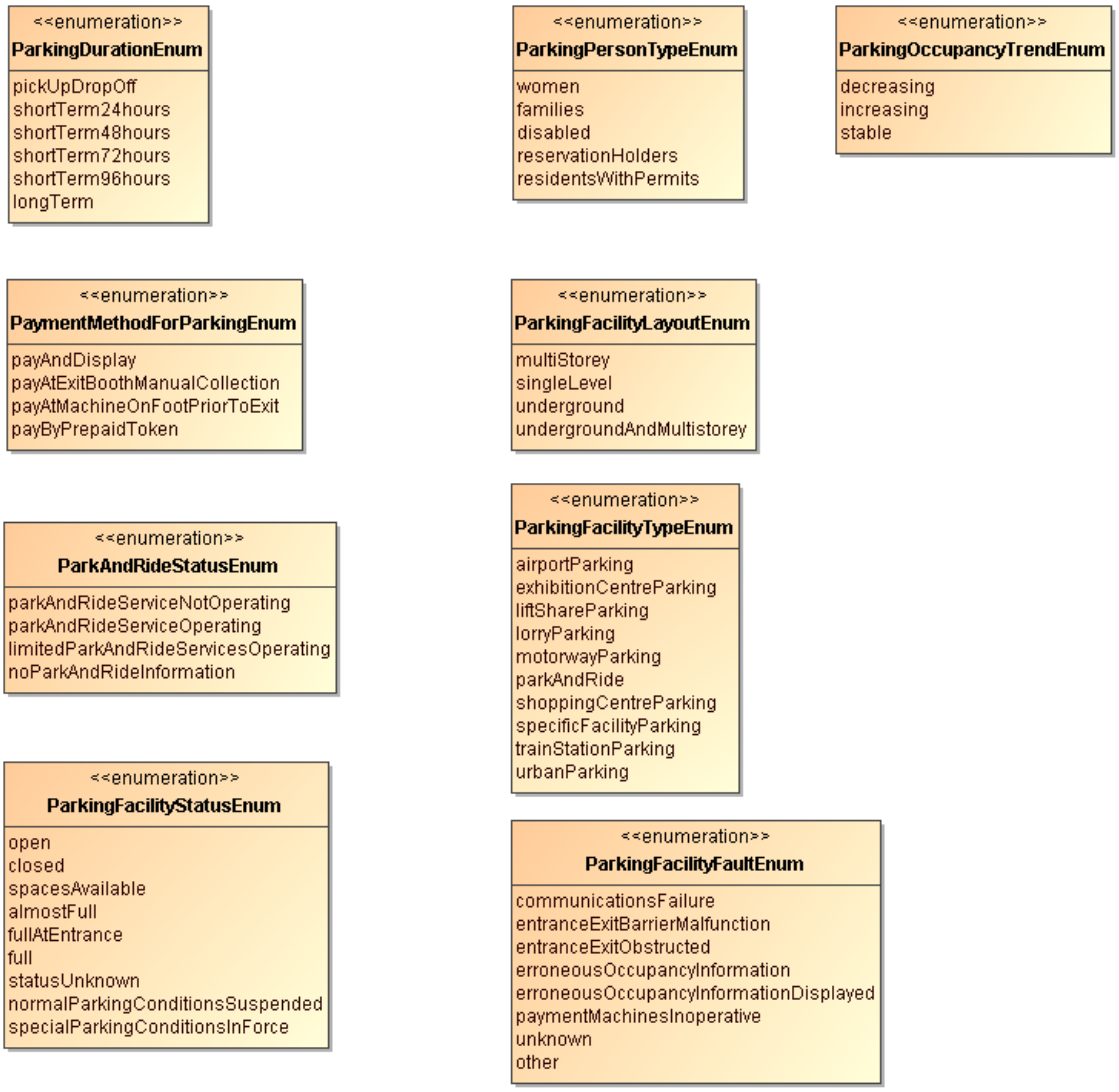
Appendix A - Datex II Class Diagrams

The diagrams in Appendix A represent the Datex II extensions to facilitate car Parks

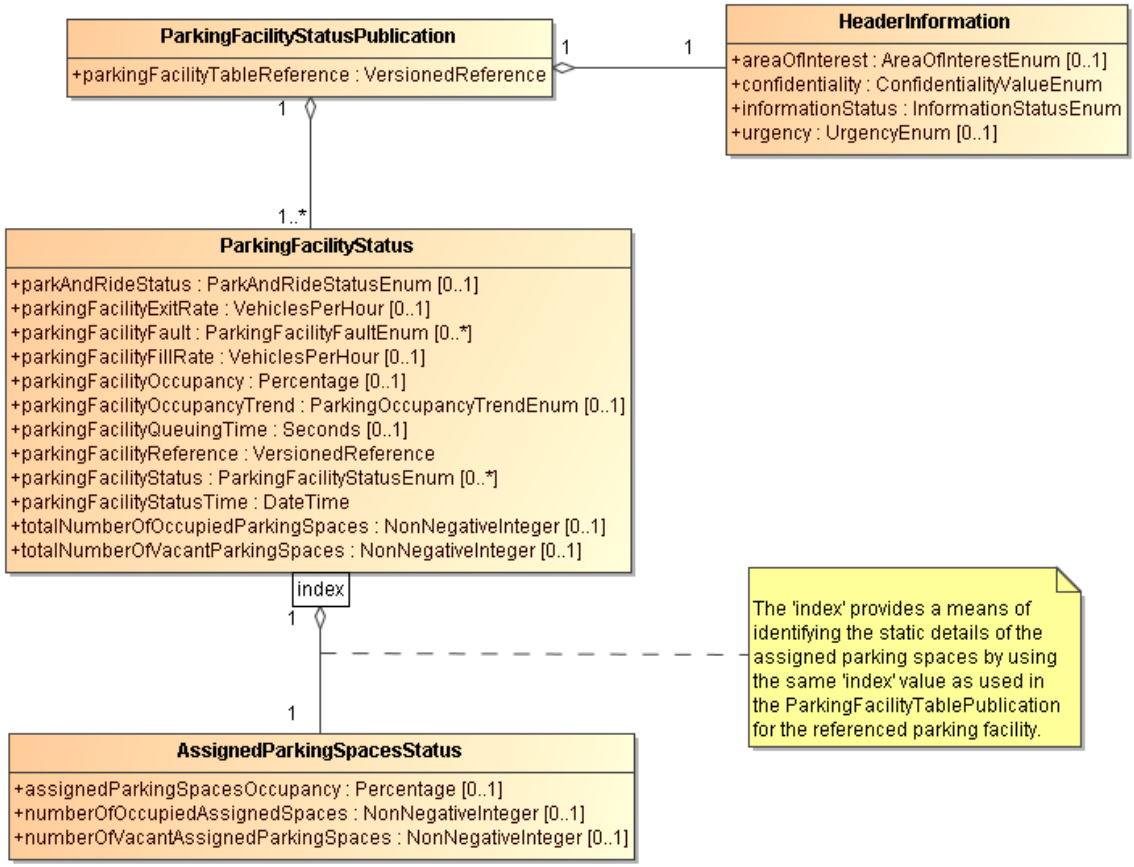
A.1 Area Extension – Class Diagram



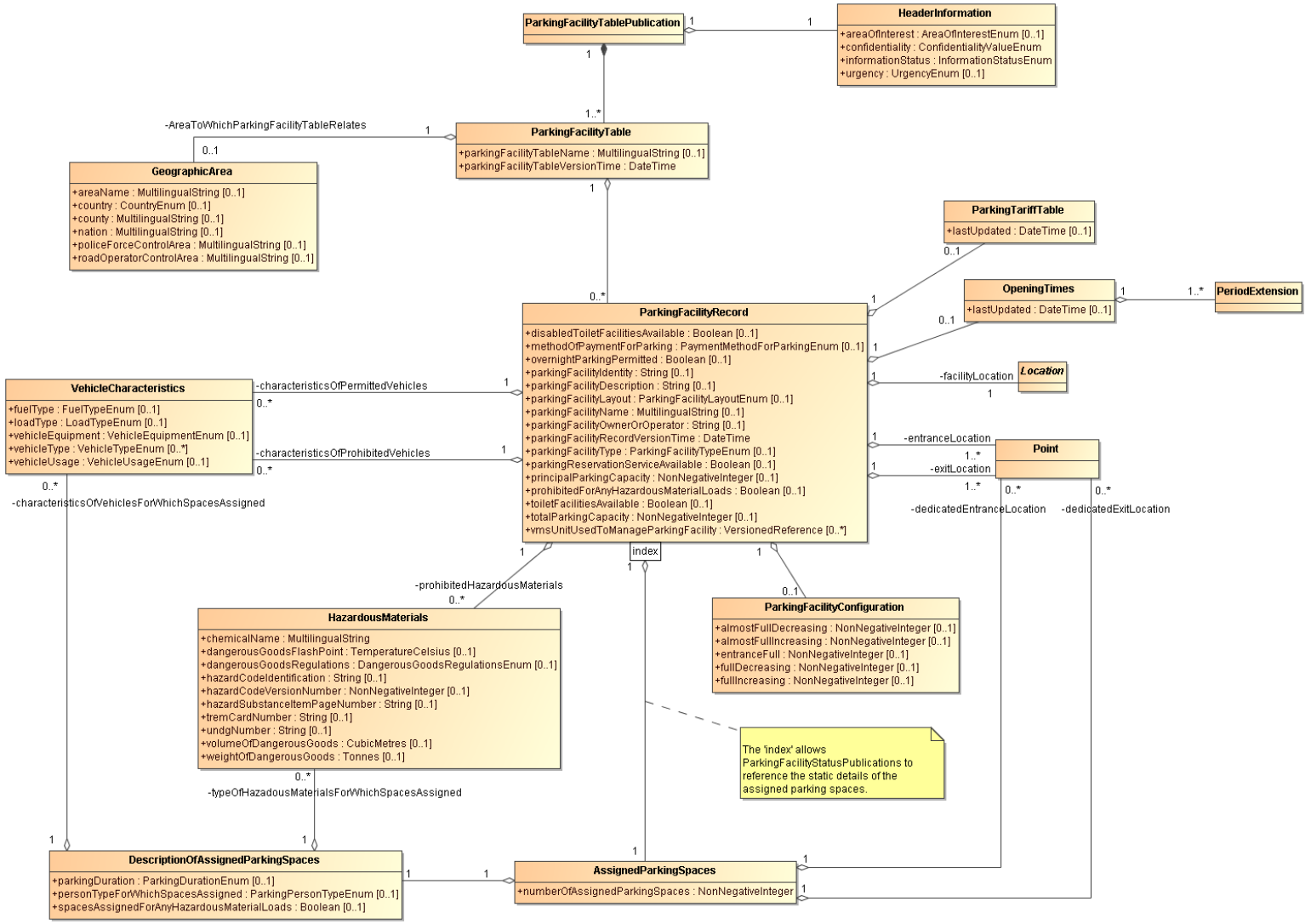
A.2 ParkingExtensionEnumerations – Class Diagram



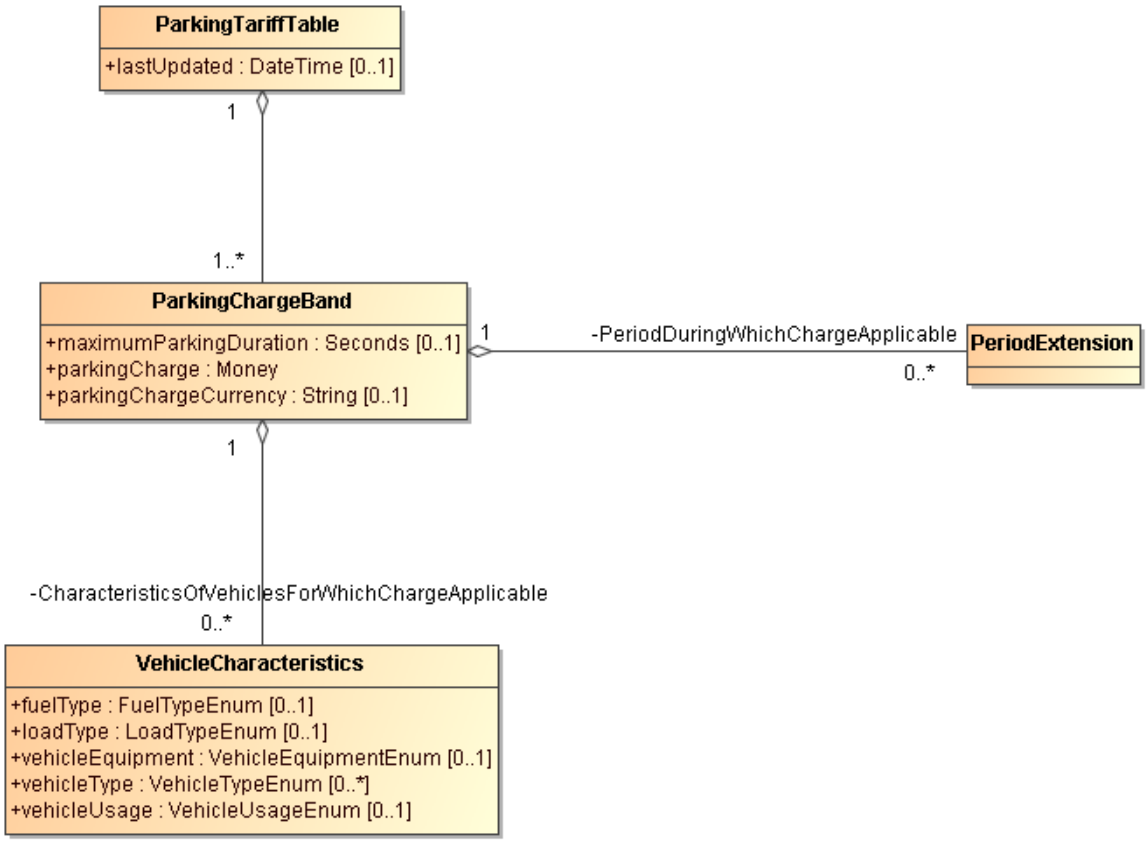
A.3 ParkingFacilityStatusPublication – Class Diagram



A.4 ParkingFacilityTablePublication – Class Diagram



A.5 ParkingTariffs – Class Diagram



A.6 PeriodExtension – Class Diagram