ROad **S**afety **ATT**ributes exchange infrastructure in **E**urope

Deliverable D4.2

Software components for data integration in digital databases

Version Number: 1.0
Produced by: Tele Atlas
Due Date: 01 07 2009
Release Date: 15 06 2010

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 – December 2010 (36 months)
EC Project Officer:	Mr. Emilio Davila-Gonzalez E-mail: emilio.davila-gonzalez@ec.europa.eu
Dissemination level:	PU

Title of document:	Software components for data integration in digital databases
Work package:	WP 4
Author(s):	Stephen T'Siobbel - Tele Atlas (editor)
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:
This document describes the (software) components for data integration in digital databases as designed and developed for the ROSATTE exchange infrastructure.

Keyword list: digital maps, data exchange, conflation, Advanced Driver Assistance Systems

Document Control Sheet

Main author(s) or editor(s): Stephen T'Siobbel - Tele Atlas (Editor)

Work area: WP 4

Document title: D4.2 – Software components for data integration in digital Databases

Version history:

Version number	Date	Main author	Summary of changes
0.1	01/03/2010	Stephen T'Siobbel Tele Atlas	First Draft – document outline
0.2	28/05/2010	Kees Wevers, Maximilian Chrzan	Addition NAVTEQ software components
0.3	01/06/2010	Stephen T'Siobbel, Marc Nause, Tele Atlas	General updates, executive summary, overview chapter
0.4	02/06/2010	Stephen T'Siobbel, Marc Nause, Tele Atlas	New illustrations, conclusions, references, terminology, edits
1.0	14/06/2010	Stephen T'Siobbel, Tele Atlas	Carry through NAVTEQs updates on reviewer recommendations

Approval:

	Name	Date
Prepared	Stephen T'Siobbel, Tele Atlas	01/06/2010
Reviewed	Anders Nilson, Triona	10/06/2010
Authorised	Jean-Charles Pandazis	15/06/2010

Circulation:

Recipient	Date of submission
ROSATTE Consortium	15/06/2010
EC	30/06/2010

Excutive Summary

This document presents a general overview of the software components which were implemented and tested by the Map Providers in the ROSATTE project with the objective to support the integration of provided safety attributes into digital map databases in a timely and correct manner. Two ROSATTE data integration services were developed by combining these software components and services in order to support the validation of the overall ROSATTE service (see WP5 results).

This project deliverable has a direct link to the deliverable D4.1 *Data Integration Methods* which sets the context of the data integration, and reviews and selects appropriate methods applicable to the general objectives of ROSATTE. The results of the validation of the data integration supported by these software components and services, an effort assessing the operation of six test sites distributed in the EU, will be described in D5.2.

The core components/services which were developed to build the overall ROSATTE data integration service includes:

- A data retrieval web service to collect safety attributes from enacting authorities/data providers,
- A Location Reference Decoding component to retrieve the spatial context of the provided safety attribute(s),
- Data processing components to map received information to map provider database,
- A feedback web service to inform update providers with the results of the data integration,
- An on-line map viewer service for the visualization of the integrated information.

In general, we may conclude that the two reference implementations of the ROSATTE data integration service as developed by the map providers in the project are very similar in terms of the overall architecture as well as in the individual components and services. The technical testing of the developed software components started in Q1 2010, early service implementations were demonstrated at the EC review in Brussels on March 2010. The validation of the service is part of work conducted in WP5.

Table of Contents

Index of Tables

Index of Figures

1. Project description

1.1. ROSATTE Contractual References

ROSATTE is a STREP submitted for the call FP7-ICT-2007-1. It 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 June 2010. It is a contract with the European Commission, DG INFSO.

The EC Project Officer is:

Emilio Davila-Gonzalez
EUROPEAN COMMISSION
DG INFSO.G.4 - ICT for Transport
Office BU31 4/50
B-1049 Brussels
Tel: +32 2 296 21 88
E-mail: Emilio.Davila-Gonzalez@ec.europa.eu

1.2. Project Objectives and scope

The ROSATTE project intends to develop the enabling infrastructure and supporting tools that will ensure European access to road safety attributes including incremental updates. This infrastructure will facilitate administrative internal functions as well as supply of data to third parties e.g. for safety relevant services.

The **overall objectives** of the project are to:

1. Facilitate access to, exchange and maintain European-wide core road safety spatial data from national/regional/local sources by standard procedures
2. Enable multi-level aggregation and update of European-wide safety map data
3. Assess the technical and organizational feasibility of this infrastructure

1.3. Key issues / Project scope

Accurate and up-to-date safety related road network attributes are particularly important for efficient road operation and administration, and for safe driving along the European road network.

For data users, the reality today is however a rather complex landscape of multiple data providers, multiple formats, varying availability and quality of data and long delays between data updates. Road authorities and infrastructure operators are usually at the beginning of the information chain, being responsible for the physical implementation, equipment and maintenance of roads.

The major problem is how to ensure timely and easy access to road information owned and maintained by thousands of road authorities. In addition mechanisms are needed to

enhance the quality of the available data in terms of accuracy, correctness and up-to-datedness, and to enable multi-level (local/national/European) aggregation of the data. With respect to a future continuous delivery and integration of updates of road attribute data, road authorities that provide such updates will be responsible for the timeliness delivery (within an agreed time period after the change of the attribute on the road), and for the correctness and positional accuracy of the data. Data integrators on their side will be responsible for correct interpretation of the received data, and correct inclusion in their digital map databases. For certain safety critical attributes, an independent certification body may be created that will be responsible for surveillance of the methods and procedures used.

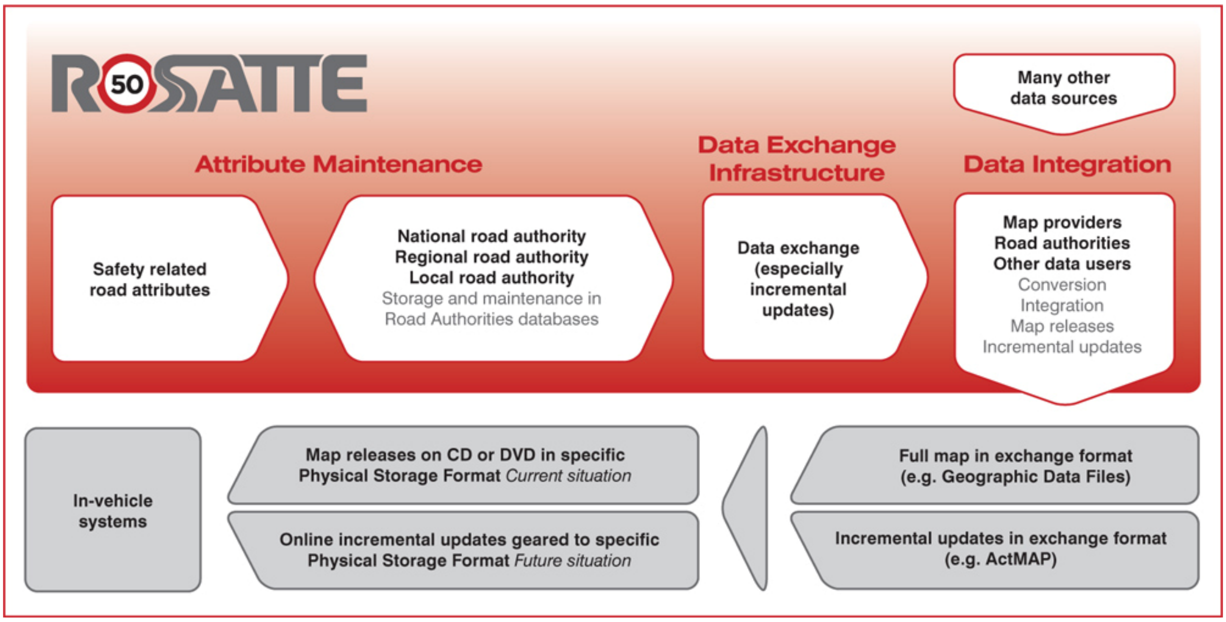

Figure 1 - The scope of the ROSATTE project

1.4. ROSATTE Project deliverables available to date

Since the start of the ROSATTE project in 2008, a number of reports have been finalized. The reports labeled for public dissemination are - or will be - available via the ROSATTE project website (www.rosatte.eu). The table below gives a comprehensive overview of available reports to date.

D1.1 State of the Art
Describes the current road authorities and infrastructure operator's situation with respect to how safety relevant data is stored, exchanged and updated.

D1.2 Requirements and Overall Architecture
Defines the project scope, user, user requirements and derived system requirements. It also gives suggestions on information model and a high-level system structure.
D2.1 Specification of tools
Conceptual specification of a road data maintenance & delivery system
D2.2 Implementations of tools
Implementations of tools for demonstration of data maintenance and access in different test beds
D3.1 Specification of Data Exchange Methods
Presents the exchange specification consisting of a data content specification, a physical exchange format specification, (to be completed) and a service specification.
D5.1 Test and Validation Plan
Describes the processes for validating the objectives of the project on each test site, tackling quality aspects, user requirements and evaluation methods.

Table 1 - Overview of project deliverables available on www.rosatte.eu to date.

1.5. Purpose of this document

The overall objective of the Data Integration work package (WP 4) of the ROSATTE project is to assess the possibilities of automatic integration of safety attributes provided by public sector into a pan-European seamless digital map database in a controllable and timely manner using the ROSATTE exchange infrastructure and exchange formats. This work package analyses, develops, tests, and validates one or more implementations for the integration of the safety attributes.

The purpose of this deliverable (D4.2) is to describe software components and services which were developed which support the integration of safety attributes for the ROSATTE project is.

2. Software components and services for ROSATTE data integration

2.1. Overview

This deliverable, D4.1 - *Data Integration Methods,* set the context of the data integration, and reviewed and selects appropriate methods applicable to the general objectives of ROSATTE. This document D4.2 – *Software Components for Data Integration* presents a general overview of all software components which were implemented and tested by the map suppliers in the ROSATTE project to support the integration of provided safety attributes into digital map databases in a timely and correct manner.

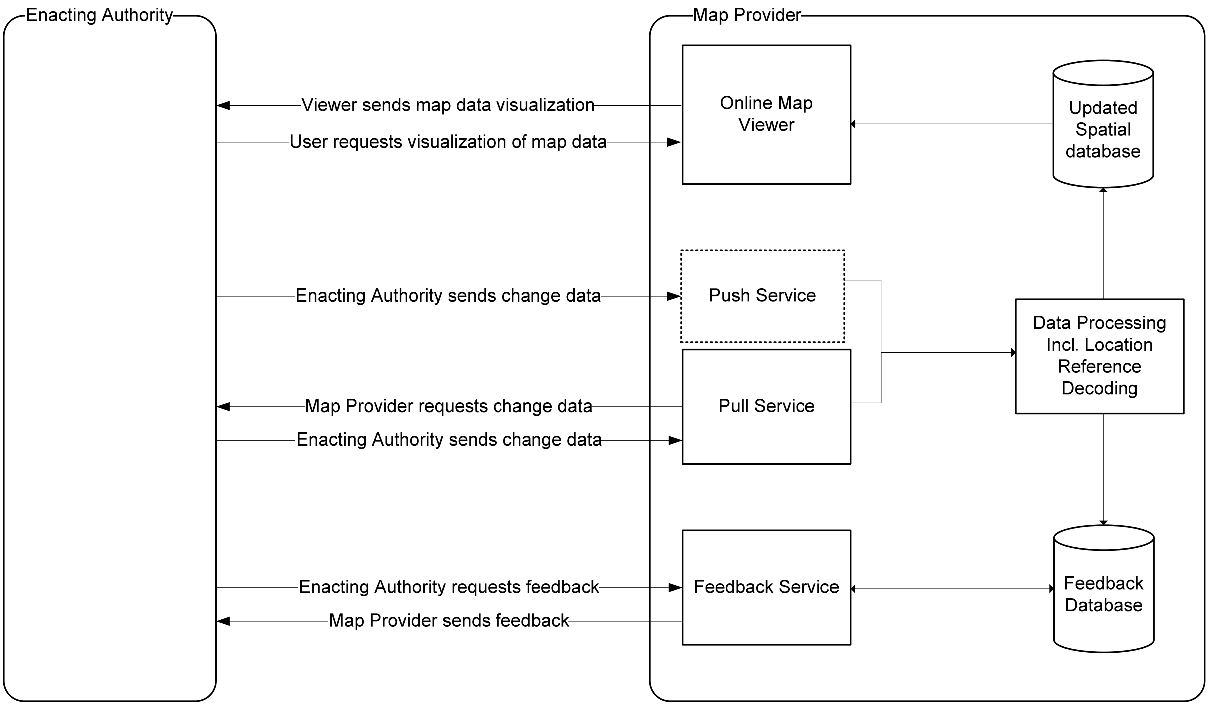

Figure 2 - ROSATTE data integration components and services.

The core components/services which were developed to build the overall ROSATTE data integration service, see Figure 2, includes a(n):

- Data retrieval web service to collect safety attributes from enacting authorities/data providers. (Remark: the "push services was described in the specifications but not implemented for test site support because of practical reasons, hence the dashed box),
- Data processing components to process received info (map it into spatial database of map provider):
    - XML validation/parsing/creation,
    - location reference decoding,
    - integrate retrieved attribute info into layer of spatial database,

    - create feedback data.
- Feedback web service to inform update providers with the results of the data integration,
- On line map viewer service for the visualization of the integrated information.

Two ROSATTE data integration services were developed by combining these software components to support the validation of the overall ROSATTE service (see WP5 results). The following chapters will describe the two implementations.

The results of the validation of the data integration supported by these software components, an effort assessing the operation of six test sites distributed in the EU, will be described in D5.4 – *Aggregated test report.*

2.2. Web Services for data exchange

The ROSATTE infrastructure specification - Deliverable D3.1 (8.2.1) - *Specification of data exchange methods* – described RESTful download services to query and download (pull) update data from the enacting authorities, as well as a feedback service which can be used to query available feedback and to download feedback referring to processed data which has been provided by the enacting authorities.

In addition to the services mentioned above, a web service to upload (push) data to the system has been implemented for testing purposes. This kind of service is also described in Deliverable D3.1 - *Specification of data exchange methods* (8.2.2). See dashed box in *Figure 2 – ROSATTE data integration components and services*

2.3. Components for data processing

In order integrate the information which is made accessible by the enacting authorities into their existing map data, the map providers had to implement several data processing components.

The data is provided by the enacting authorities in an XML format described in the ROSATTE specification - Deliverable D3.1 – *Specification of data exchange methods*. Appendix 10.2 of this document contains a definition of the XML format as XSD documents. These XSD documents allows the map providers to validate incoming data to make sure that it complies with the ROSATTE specification and can be handled by the components which handle further processing. All incoming data has to be valid in regard to the XML schema description in ROSATE.xsd.

After successful validation the received data can be parsed to retrieve the information which has been encoded in XML documents by the enacting authorities. Since the validity of the document has been proven by validation, the data from the XML document can be mapped to internal data structures representing the ROSATTE data model for further processing.

The crucial step in the processing of the data which has been retrieved from the XML documents delivered by the enacting authorities is the decoding of the locations which have been encoded using the location referencing method chosen for the ROSATTE project

(AGORA-C). Detailed information about location referencing can be found in the next chapter.

Once the locations which have been encoded by the enacting authorities have been mapped to the map providers' map data, it is possible to attach the changes of attributes provided by the enacting authorities to the corresponding map features in the map providers' data. This is done by copying the attribute information to an additional layer in a spatially enabled database which contains the map data.

After the data provided by the enacting authorities has been processed by the components described above, it is possible as a last step to provide feedback concerning the processed data to the enacting authorities. Feedback information is to be delivered in an XML format described in the ROSATTE specification - Deliverable D3.1 – *Specification of data exchange methods* (Appendix 10.2). The feedback information allows the enacting authorities to see which information has been processed and if processing of their data was successful, allowing them to detect technical problems.

2.4. Component/Service for location reference encoding/decoding

A software component able to decode AGORA-C binary codes was implemented in line with specifications. See the description of the Package "Location referencing" for the classes needed for an AGORA-C compliant location referencing - Deliverable D3.1 - *Specification of data exchange methods.*

For practical purposes, as not all ROSATTE partners providing (updates of) safety attributes were in a position to develop/use the AGORA-C location reference encoders, the Map Providers offered this functionality.

The location reference encoder service requires its user to provide the map provider's map road edge id's, the specific release version of the map, and the offsets delimiting a spatial extend of the feature. An AGORA-C location reference can be generated using a dedicated web service: see Annex 1 of one implementation.

3. ROSATTE implementations

3.1. The Tele Atlas – ROSATTE infrastructure and software tools

3.1.1. Overview

The architecture of components developed by Tele Atlas to test and validate the ROSATTE data integration is shown in**Error! Reference source not found.**. The overview represents different services (web services for data retrieval, location encoding and decoding, map rendering), processing components, and data files and spatial databases.

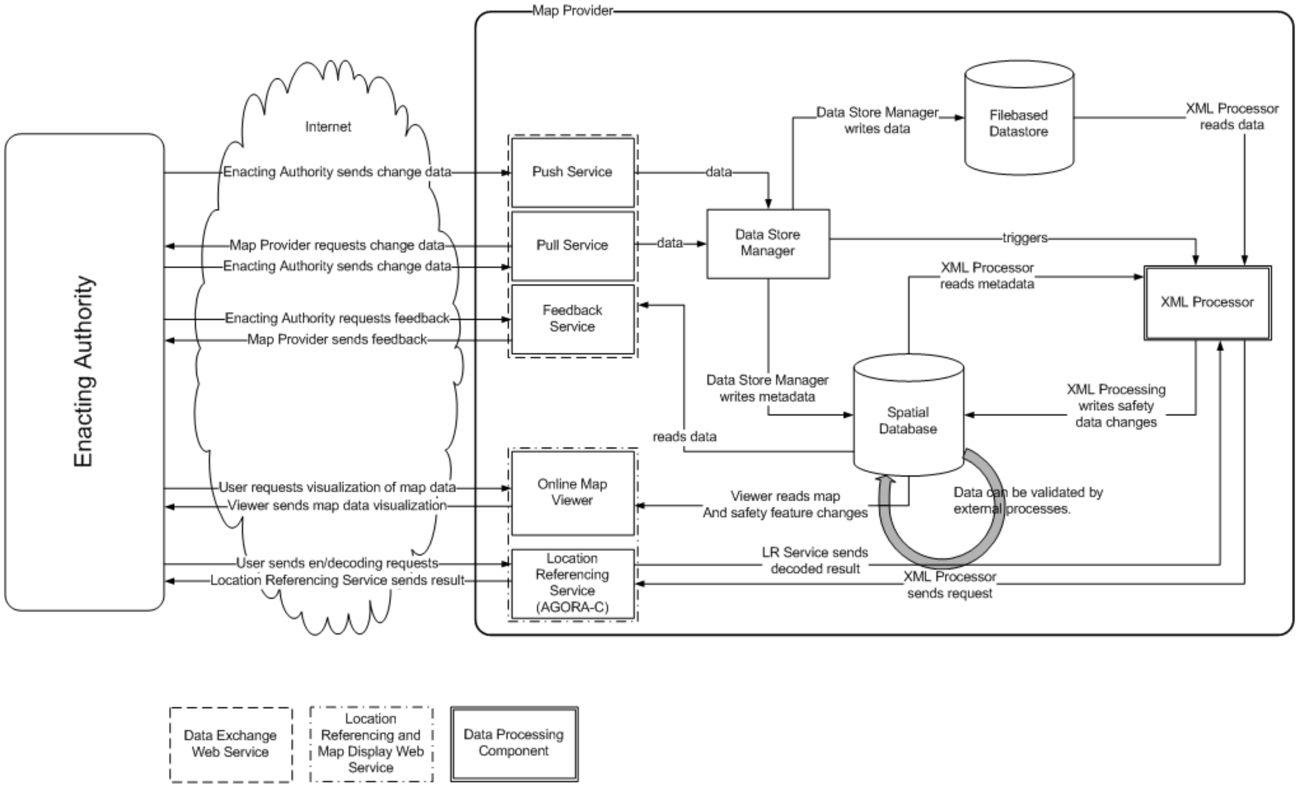

Figure 3 - Tele Atlas' ROSATTE web services and processes for data integration.

3.1.2. *Data Exchange Web Services*

Tele Atlas has implemented the RESTful download services according to Deliverable D3.1 – *Specification of data exchange methods* (8.2.1) to query and download (pull) update data from the enacting authorities, as well as a feedback service which can be used to query available feedback and to download feedback referring to processed data which has been provided by the enacting authorities (see chapter below).

In addition to the services mentioned above a web service to upload (push) data to the system has been implemented for testing purposes. This kind of service is also described in Deliverable D3.1 – *Specification of data exchange methods* (8.2.2)

Data is expected to be provided in an XML format adhering to the schema files provided by Deliverable D3.1 – *Specification of data exchange methods* (Appendix 10). All feedback from the services implemented by Tele Atlas is also provided in a format adhering to these schemas.

Data which has been added to the system is stored in a local data store in a compressed manner to be processed asynchronously to ensure availability and responsiveness of the system. Asynchronous processing also simplifies testing since the process of adding data to the system and the processing of the data can be tested independently. Testing the processing of the data can also be done without the need of causing network traffic and unnecessary load on the enacting authorities' systems.

3.1.3. *Location Referencing Web Service*

From the start of the project if became clear that a number of partners preferred to use an AGORA-C location referencing web service for encoding the locations of the safety attributes. In support, Tele Atlas developed and offered an AGORA-C web service for location referencing which adheres to ISO Standard 17572-3. In the scope of the ROSATTE project the encoding functionality of this service is used by the enacting authorities which use Tele Atlas map data. The Tele Atlas ROSATTE implementation makes use of the decoding functionality in order to decode the location data provided by all of the enacting authorities in the ROSATTE project.

Support for location references in the verbose XML location referencing format described in ISO 17572-3 has been implemented for the ROSATTE project since the Swedish/Norwegian enacting authority uses this format instead of the very compact base64 encoded binary format which is used by the other enacting authorities. In addition to these standard formats, software has been implemented to parse and process a format delivered by BALI which is similar to the standard XML location referencing format, but which differs to a degree that makes it impossible to use the standard XML location referencing parser to process this data.

3.1.4. *Data Processing Components*

Once data has been added to the Tele Atlas ROSATTE implementation, it is stored locally in a compressed form and a process is triggered by the Data Store Manager to process the

data. Depending on the workload of the system the data is processed by the XML Processor instantly or after previously started processing tasks have been finished.

During processing the update data provided by an enacting authority is read and processed one feature at a time in the order in which the features appear in the XML. The changes described for each feature are detected by parsing the XML and stored in a spatially enabled database. Then the location referencing web service described in 3.1.3 is used to find the location of the change. A geometry is created which represents the location in a Tele Atlas street map and stored in the database for visualization and further processing purposes.

In case of errors the parser creates an error message which is stored in the database for later analysis. An error can occur if the data provided as input by an enacting authority does not comply to the XML schema definition described in Deliverable D3.1 or if a location reference can not be decoded.

It is possible to receive information about the progress which has been made in processing the data which has been added to the system during runtime by querying monitoring objects which have been implemented especially for this reason.

3.1.5. Feedback and on-line MapViewer

In order to receive information about the results of the processing, a feedback service has been implemented which adheres to the specification in Deliverable D3.1 – *Specification of data exchange methods* as well as a visual means of feedback via an on-line map viewer.

The feedback web service can be queried by the enacting authorities in order to receive information if a provided dataset has already been processed and if processing was successful. The feedback also includes a GML encoded geometry which describes the location which has been decoded from the location referencing data which has been provided by the enacting authority, Figure 4 – Feedback by Map Provider on received and processed dataset (9cde05a7-2a5d-4bc4-aa79-e68c557ebeb5)and Figure 5 – Example of Feedback XML, showing GML representing spatial extent of location reference as decoded on the map of the Map Provider.

The on-line map viewer can give a visual feedback which allows a user to inspect updates and their decoded location references in relation to the Tele Atlas road network data. The decoded data is displayed as an additional layer on top of the Tele Atlas map data using geometries from the spatially enabled database. It is possible to zoom, pan and click on geometries to receive more details about the update connected to the location represented by the geometry.

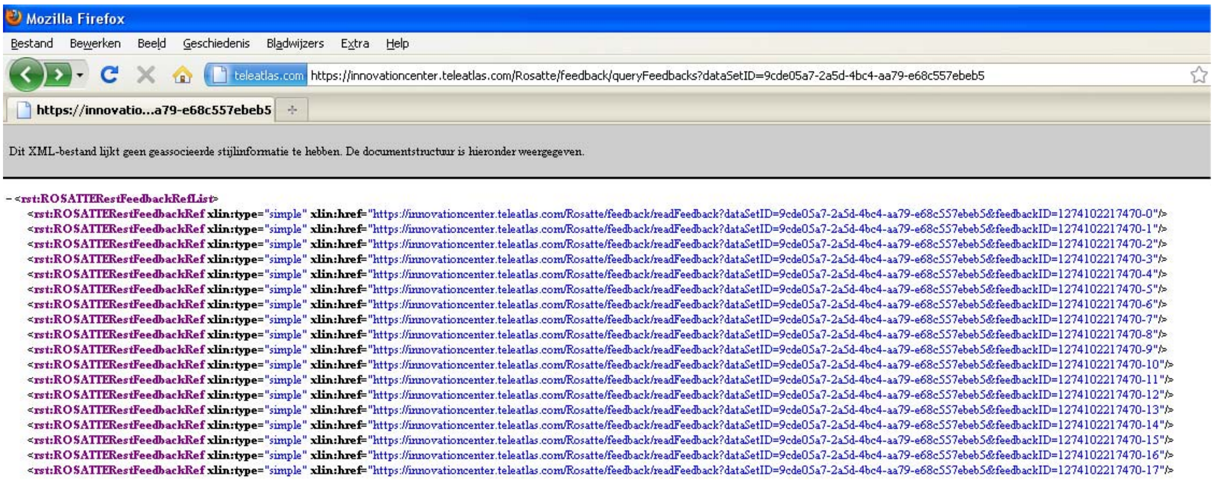

Figure 4 - Feedback by Map Provider on received and processed dataset (9cde05a7-2a5d-4bc4-aa79-e68c557ebeb5).


```
- <rst:ROSATTEFeedbackDataset>
  - <rst:feedbackInfo>
    - <rst:FeedbackInformation gml:id="i1276158935920">
      - <rst:log>
        - <rst:FeedbackLog gml:id="i1276158935800">
          - <rst:events>
              <rst:FeedbackLogEvent gml:id="i1276158935856"/>
            - <rst:FeedbackLogEvent>
              - <rst:SafetyFeatureEvent>
                - <rst:safetyFeatureId>
                  - <rst:SafetyFeatureId>
                      <rst:providerId>100264</rst:providerId>
                      <rst:id>781367b0-4f95-4934-8a09-a9c660efba7b</rst:id>
                    </rst:SafetyFeatureId>
                  </rst:safetyFeatureId>
                  <rst:eventCode>FINISHED</rst:eventCode>
                - <rst:decodedGeometry>
                  - <gml:LineString srsName="SDO:8307">
                    - <gml:coordinates decimal="." cs="," ts=" ">
                        0.4359299,47.3357443 0.4344776,47.3361994 0.4339024,47.3364088 0.4331485,47.3366834 0.4329962,47.3367388
                        0.4193941,47.3426286 0.4168353,47.343398 0.416746,47.3434248 0.4151718,47.3438496 0.4114721,47.3446564 0.
                        0.4031559,47.3456871 0.4003103,47.3458138 0.3988934,47.3458696 0.3954656,47.3460134 0.3946478,47.3460521
                        0.3786578,47.346708 0.3762556,47.3467983 0.3742905,47.3468605 0.3726452,47.3469031 0.3720807,47.3468835 0
                        0.3638386,47.3456511 0.3622592,47.3451699 0.360904,47.3446711 0.3595697,47.3440931 0.3581579,47.3433918 0
                        0.3520284,47.3385236 0.351359,47.3377331 0.350586,47.3367945 0.3501273,47.3362381 0.3500974,47.3362016 0.
                        0.3447626,47.3297618 0.3439598,47.3288503 0.3431733,47.3280571 0.3422366,47.3271497 0.3421166,47.3270334
                        0.3334617,47.3215273 0.332408,47.3210504 0.3312793,47.3206212 0.329771,47.3200471 0.3291866,47.3198159 0.
                        0.319462,47.3161144 0.3186076,47.3157826 0.316597,47.3150171 0.3159009,47.3147613 0.3136894,47.3138846 0.
                        0.3105414,47.3124349 0.3089469,47.3116221 0.3074919,47.3108126 0.3064771,47.3102377 0.3053212,47.3095835
                        0.2990878,47.3059213 0.2982033,47.3053077 0.2966048,47.304178 0.2954784,47.3033334 0.2945938,47.30262 0.2
                        0.2904803,47.2989087 0.2895633,47.2979188 0.2894642,47.2978077 0.2888641,47.2971355 0.2879113,47.2960125
                        0.2843844,47.2908051 0.2838251,47.2897508 0.2833899,47.2888643 0.2830978,47.2882746 0.2829116,47.2879169
                        0.280057,47.2828803 0.2795735,47.28228 0.2794238,47.282109 0.2792759,47.2819399 0.2792536,47.2819156
                      </gml:coordinates>
                    </gml:LineString>
                  </rst:decodedGeometry>
                </rst:SafetyFeatureEvent>
              </rst:FeedbackLogEvent>
            </rst:events>
          </rst:FeedbackLog>
        </rst:log>
        <rst:datasetIdentifier>9cde05a7-2a5d-4bc4-aa79-e68c557ebeb5</rst:datasetIdentifier>
        <rst:processed>2010-05-17T15:52:25.852+02:00</rst:processed>
      </rst:FeedbackInformation>
    </rst:feedbackInfo>
  </rst:ROSATTEFeedbackDataset>
```


Figure 5 - Example of Feedback XML, showing GML representing spatial extent of location reference as decoded on the map of the Map Provider.

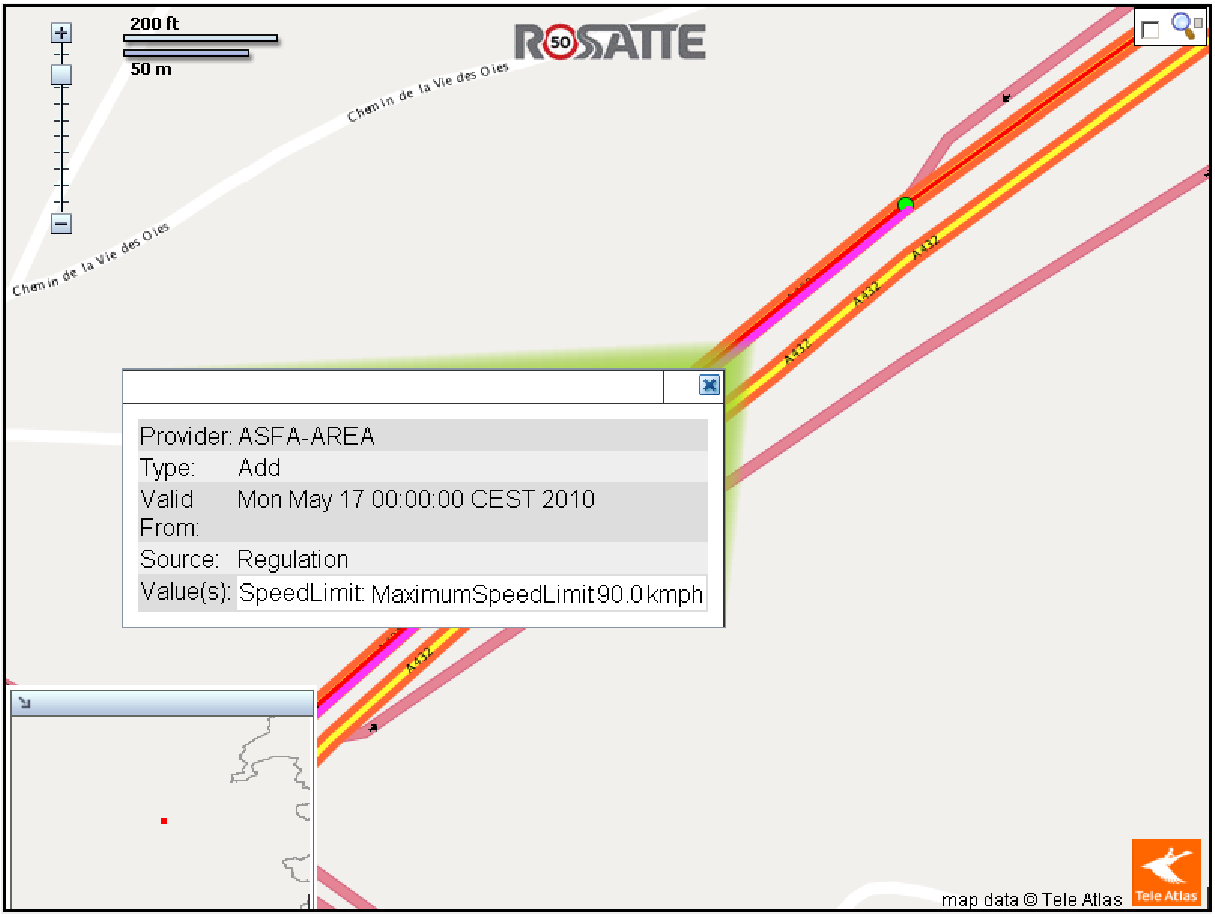

Figure 6 - Speed Limit change in the Map Viewer.

3.1.6. Implementation Platform

Tele Atlas used the Geospatial Innovation Center (IC) for the implementation to realize the ROSATTE service. The IC is a joint initiative of Tele Atlas and Oracle and kicked off in 2007. Its objectives are to:

- Establish a community of innovative LBS companies
- Provide Portal to Community to inform on services, content, pilots, ..
- Provide test bed to the community to develop efficiently innovative Geo/ LBS/ Nav applications

Thanks to the community, the array of content and service components is constantly widened and enhanced and used by an ever broadening set of applications. The Community can publish metadata on these content and service components in the Registry. Today there are over 120 Unique Community Members (Companies, Universities, Research Centers, etc.).

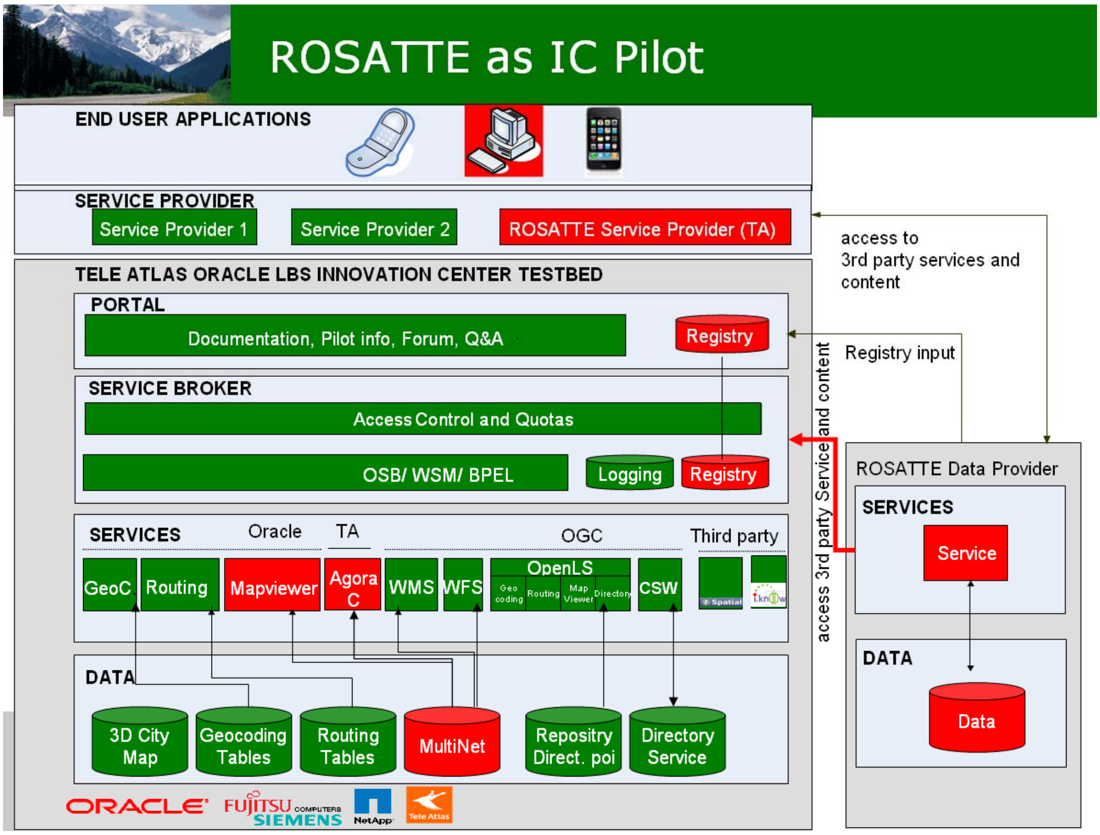

Figure 7 - Overview of the Geospacial Innovation Center. The components used by the ROSATTE software as implemented by Tele Atlas are marked in red.

The Geospatial Innovation Center currently consists of following components:

- Hardware:
  - 6 Siemens Fujitsu servers installed at Tele Atlas office in Ghent
- Software:
  - Oracle RDBMS 11g & Oracle Spatial on the DataBase Server
  - Oracle Services (Map XML request WS / response + Map Visualisation, Routing XML request / response WS, Geocoder XML request/ response WS)
  - Oracle Portal Software: BPEL, WSM, OSB, Service Registry
  - OGC & OpenLS Services (WMS (GetMap), WFS (GetFeature), OpenLS Services (Geocode, Reverse Geocode, Mapping, Routing, Directory service / yellow page)
  - Tele Atlas Geospatial Services: e.g. AGORA-C encoder-decoder, OpenLR
  - 3rd party services
- Digital map databases:

- Tele Atlas Multinet coverage Oracle 11g format on Database Server,
- 3D map data
- Etc.

3.1.7. Implementation aspects

The paragraphs below demonstrate the implemented ROSATTE data integration service. It visualizes the service status for an operator and offers some control tools. It also gives a view on the integrated data rendered on a map, the tables containing the update information and the feedback information.

3.1.7.1. Implementation technologies

Development of the Tele Atlas ROSATTE data integration service use following technologies:

- Java for the code development,
- Oracle Database with Spatial extension to store the data,
- Oracle Application Server for the code to run on,
- Oracle Mapviewer (map rendering, WMS) and Oracle Maps (JavaScript toolkit to display maps), Oracle Mapbuilder to edit the styles of displayed features,
- XML for sending and receiving messages vie web services,
- HTML for the user interface.

3.1.7.2. ROSATTE Online Map viewer

Figure 8 - On line map viewer application.

3.1.7.3. ROSATTE Process cockpit

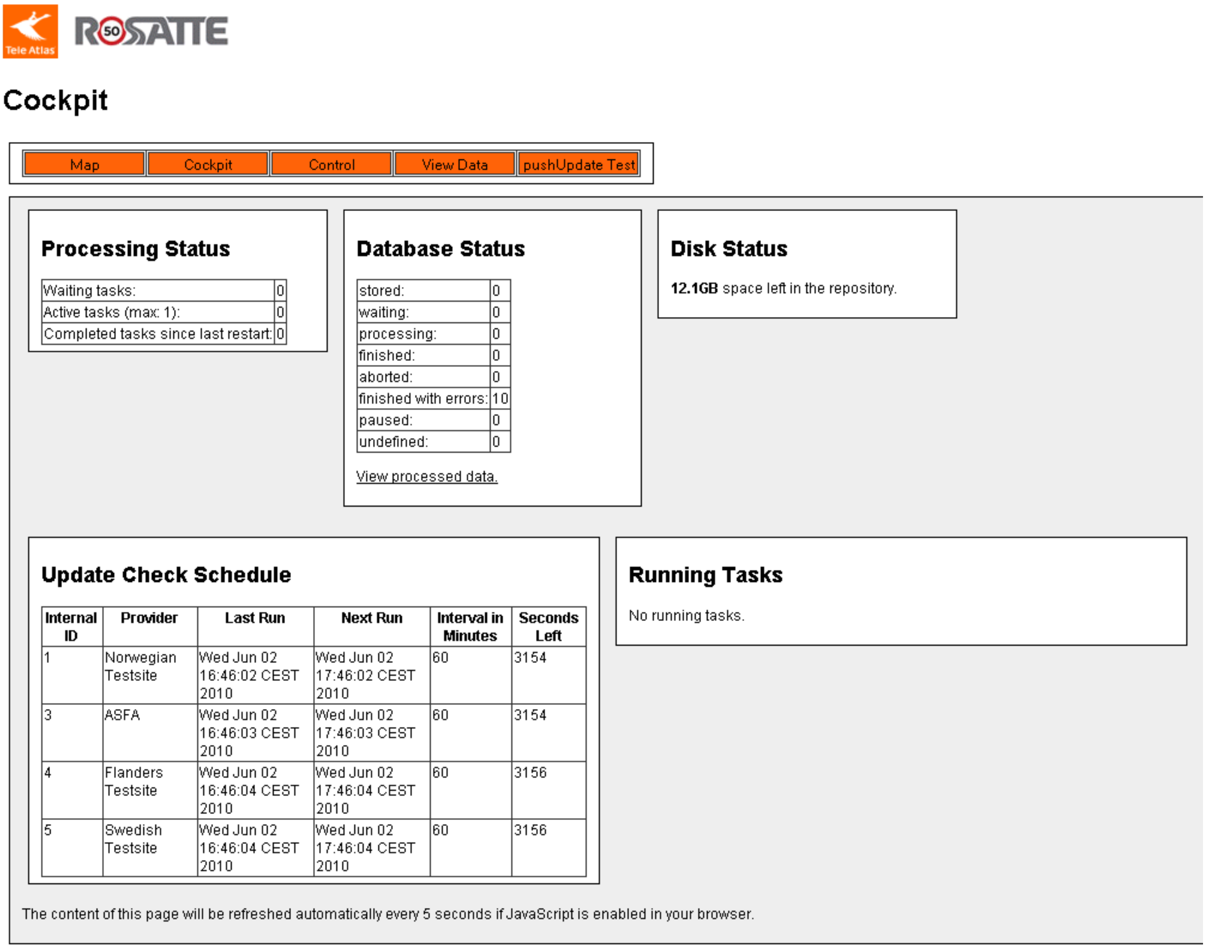

Figure 9 - ROSATTE service - Cockpit view.

3.1.7.4. ROSATTE Service Control functionality

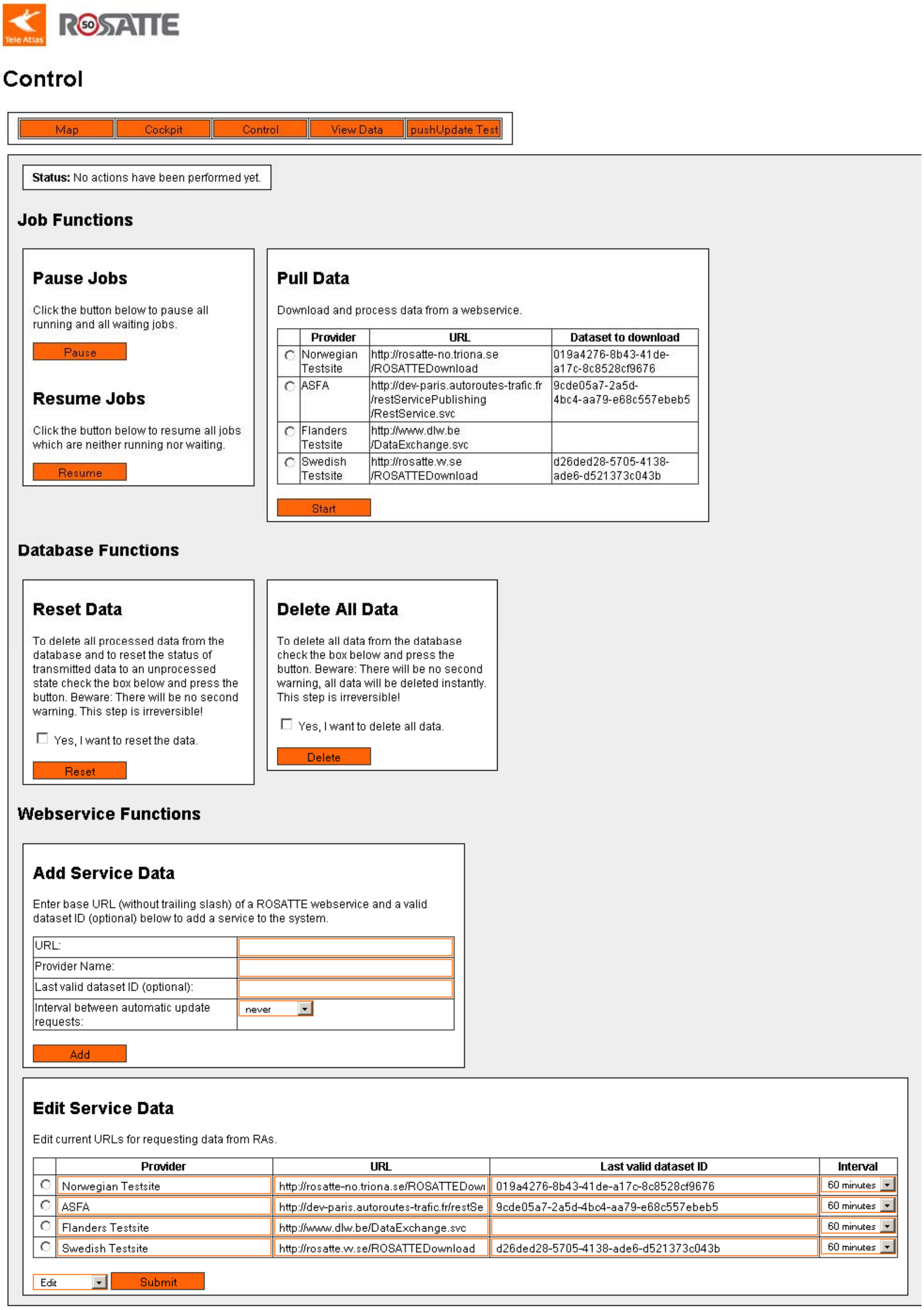

Figure 10 - ROSATTE service control functions.

3.1.7.5. On line Map viewer

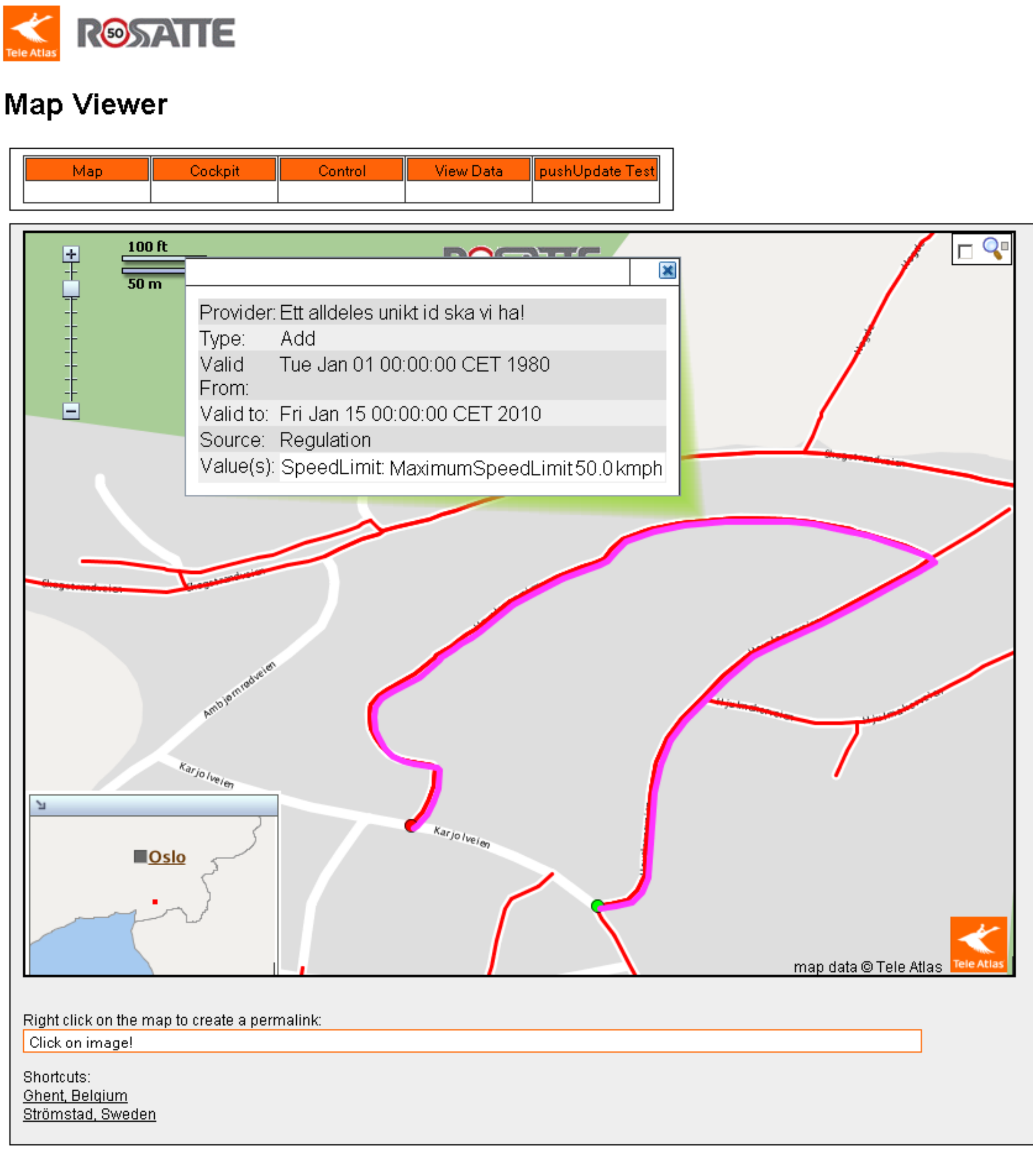

Figure 11 - The on-line map viewer showing an integrated ROSATTE update (maximum speed limit).

The Tele Atlas implementation can process safety attributes which contain location references in different formats. In addition to standard base64 encoded locations, references encoded in XML (as defined in the AGORA-C specification) and in an XML version defined by French ROSATTE partner for the French BALI and very similar to the AGORA XML, can be decoded. The decoding process matches the map agnostic location references provided by the road authorities to one road element or several connected road elements in the Tele Atlas database. Map agnostic means that AGORA-C data is independent of both the map vendor and map version. Once this step has been executed successfully, the attributes provided by the road authorities can be added to the Tele Atlas database or they can be used to update existing attributes.

In the prototype implementation data provided by the road authorities can be displayed in a map. Since the processed data is stored in a database, it is possible to easily access and use this data in additional processes in order to integrate them into products which are distributed to the end users of the Tele Atlas map data.

3.2. The NAVTEQ ROSATTE infrastructure and software tools

This section describes the components of the NAVTEQ ROSATTE infrastructure for data integration, with reference to the architecture shown in Figure 12. The overview represents different services (services for data retrieval, location encoding and decoding, map rendering), processing components, and data files and spatial databases. A main guidance for implementation is deliverable D3.1 - *Data exchange infrastructure*.

3.2.1. Data exchange services (data query)

The services consist of:

- a download pull service to query and download update data from the road authorities (also named, e.g. in the figure: enacting authorities),

- a feedback REST web service, providing road authorities the opportunity to query and download feedback resulting from the processing of update data that were provided by the road authorities.

Updates are provided by the road authorities in an XML format adhering to the schema files provided by Deliverable D3.1 1 - *Data exchange infrastructure* (Appendix 10). The NAVTEQ feedback information is provided in the same format.

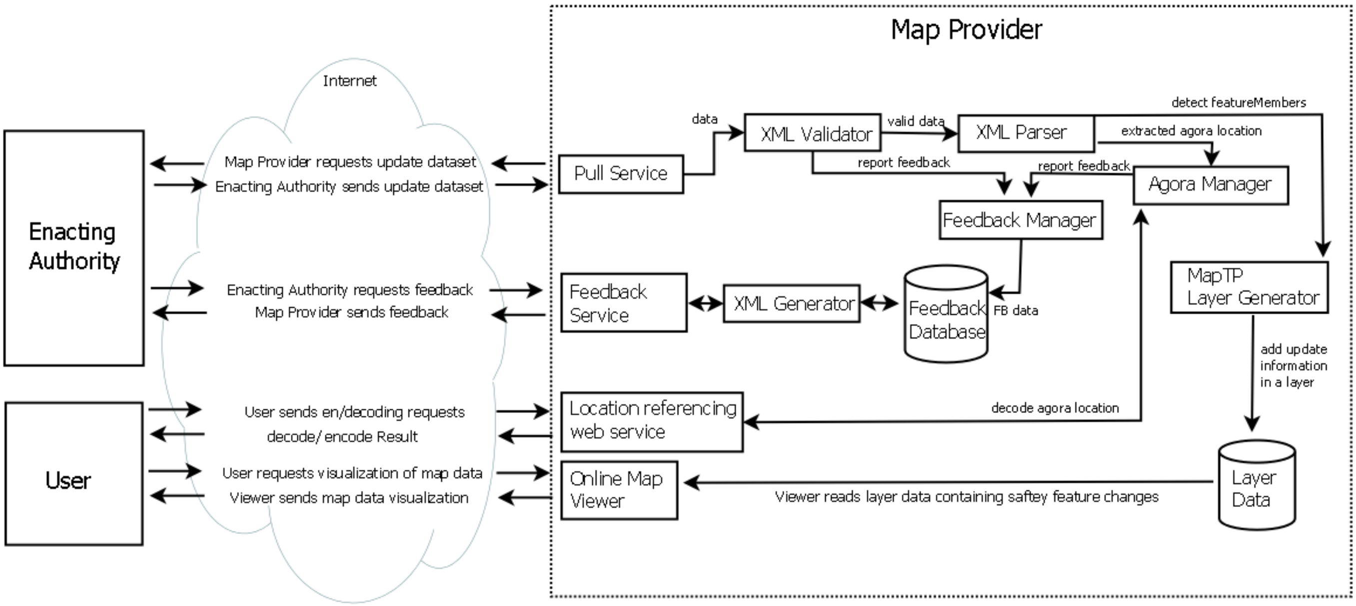

Figure 12 - Overview of NAVTEQ data integration infrastructure for ROSATTE.

3.2.2. *Location referencing web service*

The web service includes a facility for link based automatic coding and decoding from a program, and a facility with visual interface for manual encoding using the visual interface to determine the location, and for manual decoding using the visual interface for visualisation and inspection of the location. Road authority partners that use the NAVTEQ map database will use the link based web service for encoding, NAVTEQ will use the link based web service for decoding the location part of received updates. Encoding is in the compact base64 format, but as the Swedish/Norwegian road authority uses the more long-winded XML location referencing format, this is an option in the (decoding) service as well. Both formats (base64 and XML) are described in ISO Standard 17572-3. The French road authority partner uses a format similar to the XML location referencing format, but which differs to a degree that makes it impossible to use the standard XML location referencing parser to process this data. Specific software has been implemented to parse and process this format as well.

3.2.3. *Data reception and processing (mapping of safety features into the map database)*

The data are received via the data exchange Service. The data processing software reads update data from the data store, and processes these one feature at a time, in the order in which the features appear in the XML file. The location referencing web service described is used to decode the location code, and to find the location of the change. The location is then matched to a NAVTEQ database, and the update feature is stored in the database in relation to that location. The update feature information is then processed in relation to the existing map database and the features contained therein. It is possible for each change to visualise what has been changed or added. Some integrity checking on the received data has been implemented as well (inclusion in the map database). For each single update transaction that has been processed a feedback message is created, stating if the processing and implementation have been successful, and further relevant information as appropriate. A visualisation tool has been developed that enables to inspect each single processed and implemented transaction. Some statistics concerning the processed update transactions and the result of their implementation are created as well.

In case of errors an error message is created and stored to enable further analysis and inspection. Causes of error may be for instance be non-compliance with the XML schema definition as described in deliverable D3.1 - *Data exchange infrastructure* , or a location reference that cannot be properly decoded and matched. Other errors may be possible, and may be identified during development or operation, and be solved subsequently. If errors cannot be solved by improving the software implementation, they are communicated in the feedback message.

3.2.4. *Feedback service and viewer for inspection*

It is possible to query the feedback service for the following types of information: (1) if a provided dataset has already been processed; (2) the rate of success per processed data set; (3) a visualisation of the decoded geometry for each decoded location in a dataset; (4) a visualisation of the implemented change per transaction of a dataset, and a comparison with the situation in the standard NAVTEQ map database for that location and specific attribute. Additional relevant details are provided as appropriate for providing a good understanding of the update process and its results.

4. Conclusions

A high level architecture of a ROSATTE data integration service shows services for information exchange (retrieval, feedback, visualisation), components for data processing (XML parsing/creation, location reference decoding), and databases (spatial database containing the ROSATTE Safety Attributes updates, database or XML file containing feedback).

The two Map Makers in the project each developed a ROSATTE data integration service and these implementations differ only in details. Extensive technical testing was conducted during the development phase and demonstrated the technical feasibility of the service. The validation of the data integration service will be addressed in WP5.

The implementation and testing of the selected location referencing decoding component(s) requested an important amount of available resources. For the support of the test sites AGORA-C binary, AGORA-C XML, and an AGORA-C “Bali” dialect decoders were implemented, along with a AGORA-C encode web service.

5. Definitions and acronyms

5.1. Definitions

Term	Definition
Eacting Authority	Actor responsible for supervising the establishment of laws and regulations for transport and traffic. Also responsible for correct maintenance of road safety attributes.
Safety Feature / attribute	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. 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). 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. Note: 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. 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.
AGORA OpenLR	Dynamic location referencing method, which is made reference to throughout this document. 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'. AGORA was initially developed in an EU funded research project and has evolved into an ISO standard. OpenLR is an open source alternative to AGORA-C and offers free licenses.
Interface	An interface is a gateway to the functionality that a component exposes to other components or external systems.
Service	A service is a software system running on its own, not relying on user input, used by external components.
Dataset	A dataset is an identifiable collection of data.

Term	Definition
Metadata	Metadata is data about data, information making it possible to discover available data types and structures, quality parameters, geographic coverage etc., without reading the actual datasets.
Component	A component is the whole or part of a software system, seen from the outside as one unit.

5.2. Acronyms

Acronym	Definition
BALI	BAse de données des LImites de vitesse (Speed limit database, French initiative)
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. [en.wikipedia.org]
DG INFSO	Directorate-General for Information Society and Media (European Commission).
GIS	Geographic Information System is any system that captures, stores, analyzes, manages, and presents data that are linked to location. [en.wikipedia.org]
REST	Representational State Transfer
ROSATTE	Road Safety Attribute Exchange Infrastructure
UML	Unified Modelling Language [www.omg.org/uml]
WP	Work Package
XML	Extensible Markup Language: is a set of rules for encoding documents electronically. It is defined in the XML 1.0 Specification produced by the W3C and several other related specifications; all are fee-free open standards. [en.wikipedia.org]

6. References

[1] 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.

[2] BALI, see ROSATTE, deliverable D5.1 Test and validation plan. 2010. Available from: http://www.ertico.com/en/activities/safemobility/rosatte.htm

[3] ROSATTE, deliverable D3.1 Specification of data exchange methods, 2009 Available from: http://www.ertico.com/en/activities/safemobility/rosatte.htm

[4] ROSATTE, deliverable D4.1 Data Integration Methods. 2010. Available from: http://www.ertico.com/en/activities/safemobility/rosatte.htm

[5] ROSATTE, deliverable D5.4 Aggregated Test Report. 2010. Available from: http://www.ertico.com/en/activities/safemobility/rosatte.htm

Annex 1: Tele Atlas AGORA web service

INTRODUCTION

The Tele Atlas R&D AGORA-C web service was set up in order to provide the functionality to encode map locations to AGORA-C binary codes and to decode AGORA-C binary codes to map locations based on the Tele Atlas MultiNet Europe data.

ACCESS

In order to use the Tele Atlas AGORA-C web service you need to request access the service. The login data can be requested via http://www.innovation-geo-lbs.com/ (Innovation-Space -> Become user).

USAGE

The web service is available at the following URL via HTTPS POST:

https://innovationcenter.teleatlas.com/LocationTranslation/Translation

The XML structure of the request is based on the following schemas:

https://innovationcenter.teleatlas.com/LocationTranslation/schemas/ta/LocationTranslation.xsd

https://innovationcenter.teleatlas.com/LocationTranslation/schemas/ta/ADTLT.xsd

For encoding the following information are required:

- Start offset (the location of the start point on the first road segment measured as an offset between 0 (start of the segment) and 1 (end of the segment).
- End offset (the location of the end point on the last road segment measured as an offset between 0 (end of the segment) and 1 (start of the segment).
- Road segment ID (ID of a road segment based on the corresponding Tele Atlas MultiNet Oracle Release)

If you want to encode a point location first identify the road segment ID, and the offset of the point on these segment, whereas the sum of the start- and the end offset values have to be “1”. An example request for an encoded point location could look like following:

```
<adtlt:MapIdString delimiter="," startOffset="0.7" endOffset="0.3">
103800013651562
</adtlt:MapIdString>
```

In this case the point location is located at 70% (measured from the start point) of the road segment with the ID “103800013651562”.

For the encoding of a line location figure out the start offset on the first road segment, line up the next road segments, and identify the end offset on the last road segment.

In case the start of the location should represent the start point of the first road segment, it has to be set to a value of "0.0", and in case the end of the line location should be situated at the end of the last road segment, the end offset has to be defined as a value of "1.0". The following encoding request represents an example line location:

```
<adtlt:MapIdString delimiter="," startOffset="0.3" endOffset="0.7">
103800013651562,
103800013671562,
103800025826586,
-103800013550562,
103800013551562
</adtlt:MapIdString>
```

The above example describes the following situation:

- the line segment starts at the offset of 30% on the road segment with the ID "103800013651562".
- the following road segment has ID "103800013671562", and is followed by
- the road segments with the IDs "103800025826586" and "-103800013550562"
- the line location ends on the road segment with the ID "103800013551562" at an offset of 70%
- note: you can use different delimiters by definition using the "delimiter" attribute

All encoding results are in the AGORA-C physical binary format and encoded as BASE64 code to ensure a user-readable form.

MAP VERSION

A request to the AGORA-C web service allows the user to define the source map version of a list of defined IDs or the target map version used to decode an AGORA-C binary code. For encoding purposes the user has to specify this version as value of the"release" attribute (source version), which is part of the "Location" XML element. For decoding purposes the user has to use the "version" attribute (target version), which is part of the "LocationOutput" XML element (see corresponding schema definition).

Following MultiNet Oracle map releases are supported at this point:

- Europe 2008.04
- Europe 2009.02

BINARY FORMAT VERSION

The web service supports different versions of the en- and decoder physical binary format. The user is able to choose between following versions:

- "1.3", which is the pre-ISO version as defined in the Mobile.Info EU project.
- "2.4", an intermediate version.
- "3.0" as defined in the Final Draft International Standard (ISO/FDIS 17572)

The version can be specified using the "version" attribute of the "LocationReference" XML element where the format attribute is set to a value of "AGORA-C".

EXAMPLES REQUESTS

AGORA-C binary to Map ID String

```
<?xml version="1.0" encoding="UTF-8"?>
<LocationTranslationRequest
xmlns="http://www.location.com/locationtranslation"
xmlns:adtlt="http://www.location.com/adtlt"
version="1.0.0">
<LocationTranslationQuery>
<adtlt:Location>
<adtlt:LocationReference format="AGORA-C"
version="3.0">AUEBMAA9AiAGBBcWxC4HFyklIU6iSA7sQAUDBEsyMDHCQAQKCdEExO
LNAAUCBAQTEtEO8AHAYM1ABQMEBEsyMDEGQA==</adtlt:LocationReference>
</adtlt:Location>
<TranslationOutput format="MAP-ID-STRING" version="2009.02"/>
</LocationTranslationQuery>
</LocationTranslationRequest>
```

Map ID String to AGORA-C binary

(Please remove the line break in the line with the Map IDs after copying this code.)

```
<?xml version="1.0" encoding="UTF-8"?>
<LocationTranslationRequest
xmlns="http://www.location.com/locationtranslation"
xmlns:adtlt="http://www.location.com/adtlt"
version="1.0.0">
<LocationTranslationQuery>
<adtlt:Location xmlns:adtlt="http://www.location.com/adtlt"
release="2009.02">
<adtlt:MapIdString delimiter="," startOffset="0.3"
endOffset="0.7">103800013651562,103800013671562,103800025826586,
-103800013550562,103800013551562</adtlt:MapIdString>
</adtlt:Location>
<TranslationOutput format="AGORA-C" version="3.0"/>
</LocationTranslationQuery>
</LocationTranslationRequest>
```

https://innovationcenter.teleatlas.com/LocationTranslation/schemas/ta/ADTLT.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns:gml="http://www.opengis.net/gml"
xmlns:adtlt="http://www.location.com/adtlt"
xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.location.com/adtlt"
elementFormDefault="qualified" version="1.4">
    <import namespace="http://www.opengis.net/gml"
schemaLocation="../gml/2.1.2/geometry.xsd"/>
    <!-- ############ Abstract datatypes for location
translation############ -->
    <element name="TranslationOutput" type="adtlt:TranslationOutputType"/>
    <complexType name="TranslationOutputType">
        <attribute name="format" type="adtlt:TranslationOutputFormatType"
use="required"/>
        <attribute name="version" type="string" use="required"/>
    </complexType>
    <element name="Location" type="adtlt:LocationType"/>
    <complexType name="LocationType">
        <choice>
            <element ref="gml:LineString"/>
            <element ref="adtlt:MapIdString"/>
            <element ref="adtlt:LocationReference"/>
            <element ref="adtlt:TMCLocation"/>
            <element name="GenericFormat" type="string"/>
        </choice>
        <attribute name="locationQuality" use="optional">
            <simpleType>
                <restriction base="double">
                    <minInclusive value="0"/>
                    <maxInclusive value="1"/>
                </restriction>
            </simpleType>
        </attribute>
        <attribute name="release" type="string" use="optional"/>
    </complexType>
    <element name="LocationReference" type="adtlt:LocationReferenceType"/>
    <complexType name="LocationReferenceType">
        <simpleContent>
            <extension base="base64Binary">
                <attribute name="format"
type="adtlt:LocationReferenceFormatType" use="required"/>
                <attribute name="version" type="string" use="required"/>
            </extension>
        </simpleContent>
    </complexType>
    <element name="MapIdString" type="adtlt:MapIdStringType"/>
    <complexType name="MapIdStringType">
        <simpleContent>
            <extension base="string">
                <attribute name="delimiter" type="token" use="optional"
default=" "/>
                <attribute name="startOffset" use="optional" default="0.0">
                    <simpleType>
                        <restriction base="float">
```

```
                            <minInclusive value="0.0"/>
                            <maxInclusive value="1.0"/>
                        </restriction>
                    </simpleType>
                </attribute>
                <attribute name="endOffset" use="optional" default="0.0">
                    <simpleType>
                        <restriction base="float">
                            <minInclusive value="0.0"/>
                            <maxInclusive value="1.0"/>
                        </restriction>
                    </simpleType>
                </attribute>
            </extension>
        </simpleContent>
    </complexType>
    <simpleType name="TranslationOutputFormatType">
        <restriction base="string">
            <enumeration value="AGORA-C"/>
            <enumeration value="OPENLR"/>
            <enumeration value="MAP-ID-STRING"/>
            <enumeration value="GML-LINESTRING"/>
            <enumeration value="GENERIC-FORMAT"/>
        </restriction>
    </simpleType>
    <simpleType name="LocationReferenceFormatType">
        <restriction base="string">
            <enumeration value="AGORA-C"/>
            <enumeration value="OPENLR"/>
        </restriction>
    </simpleType>
    <simpleType name="MapVendorType">
        <restriction base="string">
            <enumeration value="NAVTEQ"/>
            <enumeration value="TELE ATLAS"/>
            <enumeration value="SRA"/>
            <enumeration value="OBB"/>
        </restriction>
    </simpleType>
    <simpleType name="ResponseType">
        <restriction base="string">
            <enumeration value="SUCCESSFUL"/>
            <enumeration value="UNSUCCESSFUL"/>
        </restriction>
    </simpleType>
    <element name="TMCLocation" type="adtlt:TMCLocationType"/>
    <complexType name="TMCLocationType">
        <attribute name="delimiter" type="token"/>
        <attribute name="tmctableversion" type="string"/>
    </complexType>
    <simpleType name="Error">
        <restriction base="string"/>
    </simpleType>
</schema>
```