LIFE+10 ENV/IT/000389

INTEGREEN

Action 3: System design

D.3.1.2

Vehicle-to-centre front-end and web interfaces design

Project Coordinating Beneficiary	Municipality of Bolzano
Project Associated Beneficiary n.2	TIS innovation park (TIS)
Project Associated Beneficiary n.3	Austrian Institute of Technology (AIT)
Project Associated Beneficiary n.4	Österreichisches Forschungs- und Prüfzentral Arsenal (ÖFPZ)

Document history

Date	Document Author(s)	Document Contribution
31/03/14	Roberto Cavaliere, Paolo Valleri, Stefano Seppi (TIS), Reinhard Kloibhofer, Wolfgang Ponweiser (AIT)	Document finalization

Dissemination level: CO[1]
Delivery month: M31
Status: draft

[1] **PU** = Public; **CO** = Confidential (for members of the consortium only, including the European Commission); **RE** = Restricted, i.e. only a selected target of stakeholders defined by project consortium and validated by European Commission can access to the document. The final dissemination level will be defined in cooperation between the INTEGREEN consortium and the European Commission.

Table of Contents

Table of Figures

Table of Tables

1 Introduction

1.1 Purpose of the document

The main objective of Task 3.1 Action n.3 is to obtain a complete technical definition of the new Environmental Supervisor Centre within the proposed INTEGREEN architecture (**Fehler! Ungültiger Eigenverweis auf Textmarke.**). This document covers in a specific way the vehicle data-source and the entire front-ends layer (in the original denomination named as "*vehicle-to-centre front end*" and "*public web interfaces*"); all other components foreseen at the traffic management centre layer are reported in the deliverable D.3.1.1 [1]. The work on the prototype mobile system in on the contrary covered by the activities of Task 3.2, which is reported in deliverables D.3.2.1 and D.3.2.2 [2]- [3], respectively. This document aims to represent a specific plan for its following actuation, which is foreseen in Action n.4.

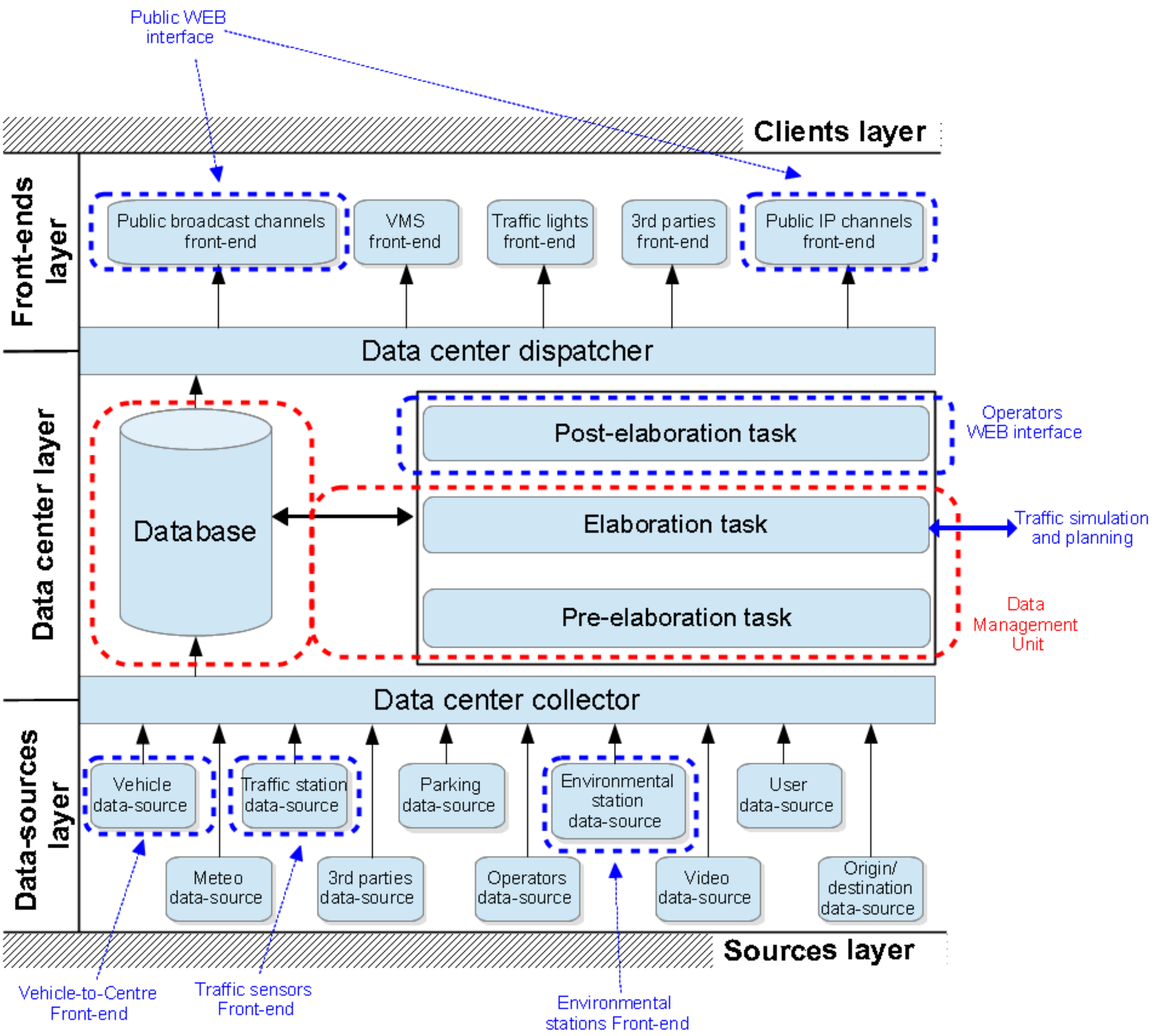

Figure 1: INTEGREEN system architecture [4].

The functional and logical architecture of the single components and of the system as a whole will be outlined. Hardware and software components will be selected and where necessary designed ad hoc, and communication specifications, protocols and data models are clearly specified.

1.2 Specification definition methodology

The design process is the second step of the V-model approach followed within the project (Figure 2). This approach is a very common engineering methodology which is very often applied for the deployment of ITS projects like INTEGREEN, as stated in a very comprehensive handbook provided by the U.S. Department of Transportation – Federal Highway Administration [5]. More information about the methodology followed in this project phase can be found in Chapter 1 of D.3.1.1.

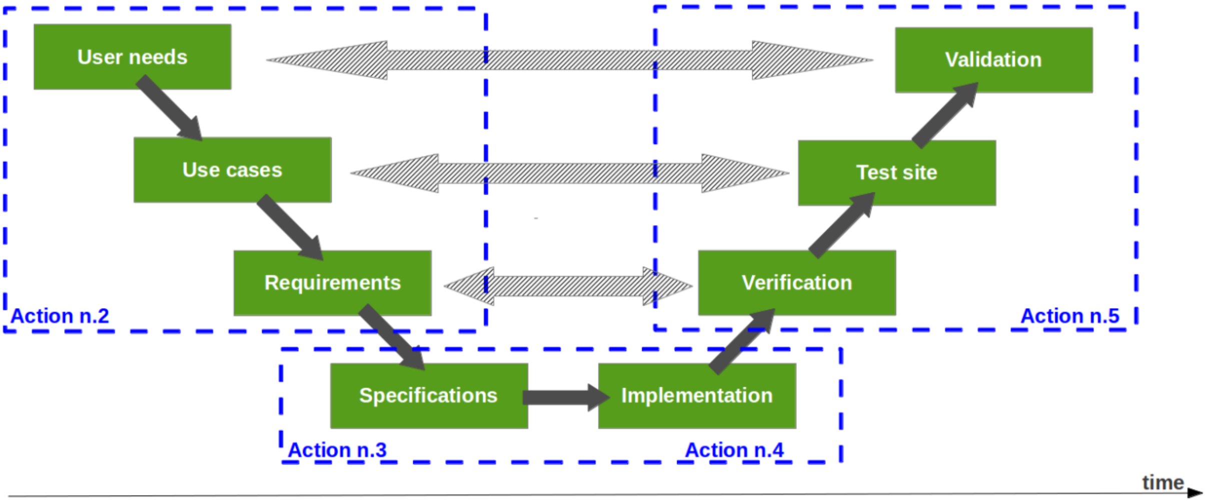

Figure 2: The V-model approach adopted in INTEGREEN [4].

1.3 Document structure

This deliverable is organized as follows. Chapter 2 presents the design choices defined for the interface between vehicle front-end and vehicle data source, and Chapter 3 presents the structure that has been introduced at the front-end layer of the Environmental Supervisor Centre, where relevant data and information are distributed to a multitude of potential service providers.

2 Vehicle Front-End / Data Source interface design

During the final stage of Action n.2 project beneficiaries, and in particular TIS (responsible for the design and implementation of the Supervisor Centre) and AIT (responsible for the design and implementation of the INTEGREEN mobile system), have decided to introduce a reference scalable architecture for the real-time data delivery between the mobile probes (i.e. the vehicles) and the central collection point of the INTEGREEN system. The idea is to cover this chain in three different consecutive phases:

- **phase 1**: the on-board telematic unit collects all the raw measurements performed by the set of sensors placed in the on-board monitoring units (traffic and environmental);
- **phase 2**: the on-board telematics unit delivers the pre-processed measurements to the vehicle front-end through its communication interface. The vehicle front-end is a server side intermediate entity which is responsible of the remote storage of the data collected by all prototype mobile systems of INTEGREEN; in this first implementation phase, this entity is going to be under the responsibility of AIT;
- **phase 3**: the vehicle data source synchronizes with the vehicle front-end in order to receive an update of the data collected by the INTEGREEN mobile probes. Similarly to the other data sources of the Supervisor Centre, the vehicle data source is a simple client entity which acts as a "end-point" for external actors that can provide relevant data to the INTEGREEN system.

Why this kind of architecture? Because in this way it will be quite simple to add distributed data coming from other 3rd parties vehicles' fleets, which will probably have an *Automatic Vehicle Monitoring* (AVM) system capable to remotely receive relevant information from their vehicles, typically *Floating Car Data* (FCD) as described in Chapter 2 of D.2.2.1 [6]. As a matter of example, in INTEGREEN a specific adapter will be introduced at the vehicle data source level capable to receive on a real-time basis the position of the buses driving in the urban area of Bolzano that are managed by the local public transport operator SASA (Figure 3). On road links where no fast tracks are located, this data can provide very useful information to understand the level of traffic congestion in the urban road network, and direct comparisons will be possible with the traffic data provided by the mobile systems or by other fixed monitoring stations (e.g. O/D Bluetooth detectors).The specification work for this chain has mainly covered two aspects:

- the specification of the **data package** that each mobile probe must provide to the vehicle front-end and successively to the vehicle data source;
- the specification of how the **interface between vehicle front-end and vehicle data-source**.

It is worth noting that the scope of this document is primarily on phase 3 of the communication chain; the design choices for phase 1 and 2 are already presented in D.3.2.1 [2].

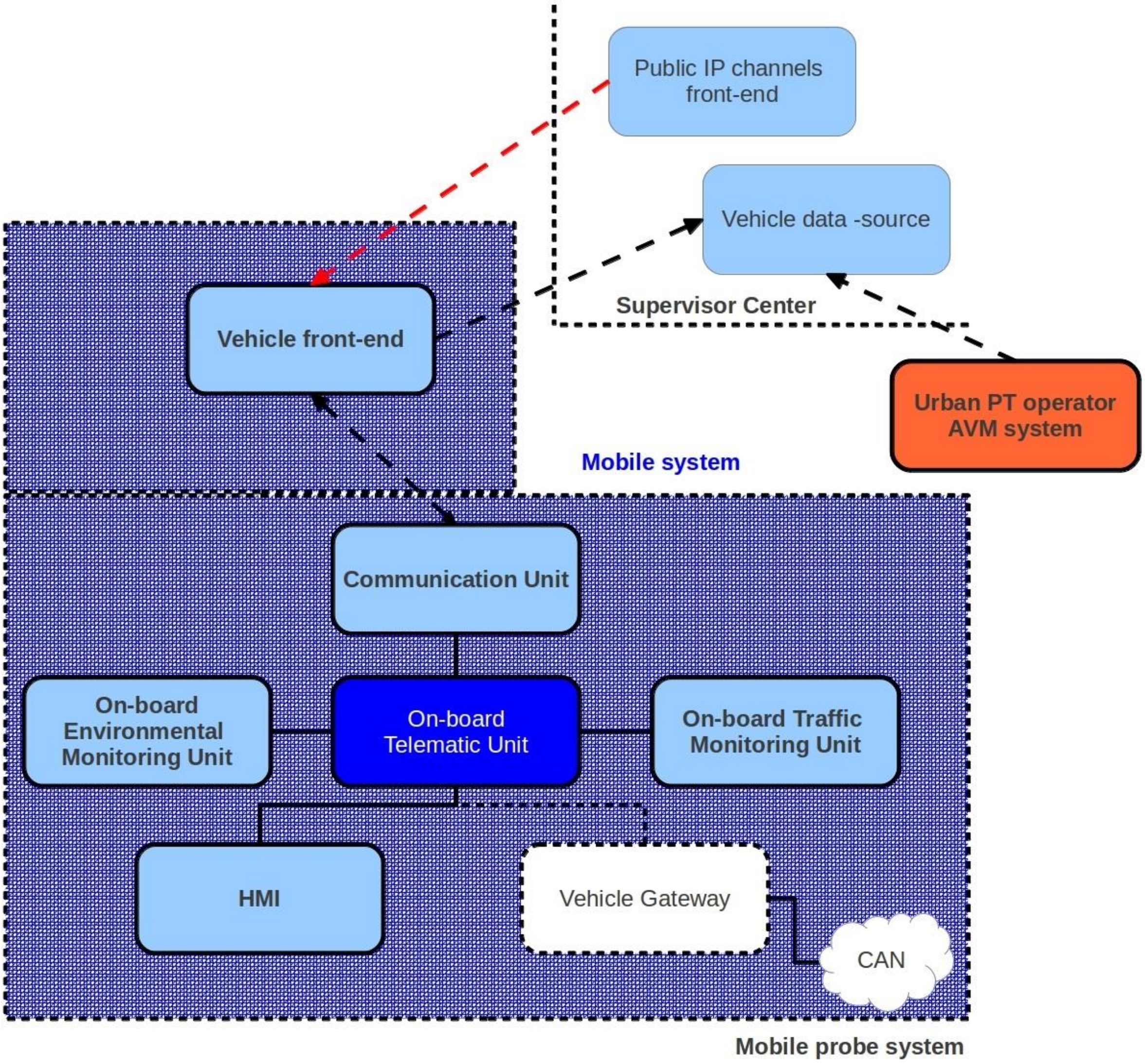

Figure 3: The reference architecture for the INTEGREEN mobile system.

2.1 Data package specification

Being the INTEGREEN mobile system a first prototype of “mobile probe”, and one of the first ones available at the state-of-art as far as mobile air pollution monitoring through low cost sensors is concerned, the list of parameters that are covered in this first deployment is quite long and complex. In fact, the field operational tests of INTEGREEN could be a great opportunity to investigate a lot of aspects that are relevant for the research community, e.g. for *naturalistic driving studies* (NDS). The list of data that each complete mobile system is presented in the following tables. It is worth noting how the specification of this package could be reduced to a relevant subset only, or further extended depending on future applications. All mobile systems developed in the INTEGREEN project will be however in the condition to provide this full dataset.

- **no2_1**: NO_2 Alphasense sensor;
- **no2_2**: NO_2 e2v sensor;

- **o3_1**: O_3 Alphasense sensor;
- **co_1**: CO e2v sensor;
- **res_1, res_2**: generic additional air pollution sensors;
- **temp_1**: air temperature sensor near air pollution detectors;
- **rh_1**: relative humidity sensor near air pollution detectors;
- **af_1**: air flow meter in correspondence of air pollution detectors;
- **gps_1**: GPS data package provided by the GPS receiver (positioned in the on-board telematics unit);
- **system_id**: identification data related to the mobile probe delivering the data;
- **can_1**: the reference subset of data read from the CAN-bus network of the vehicle;
- **imu_1**: the cinematic data provided by the Inertial Measurement Unit of the on-board traffic monitoring unit.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
no2_1_ppb	82	signed Real	0.0 … 100000.0	Measured NO_2 concentration [ppb], values rounded to 1 ppb, for tests also resolution 0.1 ppb possible
no2_1_runtime_s	8569	unsigned Integer	32 bit	Time since last power-on of sensor [s] , zero means runtime unknown
no2_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 1: no2_1: output of the NO_2 Alphasense sensor.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
no2_2_ppb	84	signed Real	0.0 … 100000.0	Measured NO2 concentration [ppb], values rounded to 1 ppb, for tests also resolution 0.1 ppb possible
no2_2_runtime_s	8569	unsigned Integer	32 bit	Time since last power-on of sensor [s] , zero means runtime unknown
no2_2_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 2: no2_2: output of the NO_2 e2v sensor.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
o3_1_ppb	75	signed Real	0.0 … 100000.0	Measured O_3 concentration [ppb], values rounded to 1 ppb, for tests also resolution 0.1 ppb possible
o3_1_runtime_s	8569	unsigned Integer	32 bit	Time since last power-on of sensor [s] , zero means runtime unknown
o3_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 3: o3_1: output of the O_3 Alphasense sensor

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
co_1_ppm	75	signed Real	0.0 … 100000.0	Measured CO concentration [ppm], for tests also resolution 0.1 ppm possible
co_1_runtime_s	8569	unsigned Integer	32 bit	Time since last power-on of sensor [s] , zero means runtime unknown
co_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 4: co_1: output of the CO e2v sensor.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
res_1_a	84	signed Real	0.0 … 100000.0	Measured of a first future additional sensor
res_1_runtime_s	8569	unsigned Integer	32 bit	Time since last power-on of sensor [s] , zero means runtime unknown
res_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 5: res_1: output of first future additional sensor.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
res_2_a	84	signed Real	0.0 … 100000.0	Measured of a second future additional sensor
res_2_runtime_s	8569	unsigned Integer	32 bit	Time since last power-on of sensor [s] , zero means runtime unknown
res_2_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 6: res_2: output of future additional sensor.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned	64 bit	Timestamp [ms]

		Integer		
temp_1_c	23.2	signed Real	-40.00 … 120.00	Air temperature [°C], values rounded to 0.1 °C, for tests also resolution 0.01 °C possible
temp_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 7: temp_1: output of the air temperature sensor near gas sensors.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
rh_1_pct	47.3	signed Real	0.00 … 100.00	Relative humidity [%], values rounded to 0.1 %, for tests also resolution 0.01 % possible
rh_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 8: rh_1: output of the relative humidity sensor near gas sensors.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
af_1_sccm	150	signed Real	0 … 10000	Air flow [sccm], values rounded to 1 sccm, for tests also resolution 0.1 sccm possible
af_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 9: af_1: output of the airflow sensor near the air pump.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
gps_1_long_deg	16.244373	signed Real	-179.999999… 180.000000	GPS longitudinal position in degree referred to WGS84, positive values means east, negative values means west
gps_1_lat_deg	48.343838	signed Real	-90.000000 … 90.000000	GPS latitude position in degree referred to WGS84, positive values means north, negative values means south
gps_1_alt_m	125.3	signed Real	0.0 … 10000.0	GPS altitude [m]
gps_1_speed_mps	36.2	unsigned Real	0.0 100.0	GPS horizontal speed [m/s]
gps_1_hdg_deg	184.70	unsigned Real	0.0 359.99	GPS heading [degree]
gps_1_sat_nr	7	unsigned Integer	1 … 12	Number of used GPS satellites
gps_1_pdop_nr	1.8	unsigned Real		GPS Position Dilution of Precision
gps_1_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 10: gps_1: output of the GPS sensor in the telematic unit.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
id_vehicle_nr	50	unsigned Integer	1 … 1000000	Vehicle ID, zero means ID unknown
id_system_nr	4	unsigned Integer	1 … 1000	Mobile System ID, zero means ID unknown
id_driver_nr	69	unsigned Integer	1 … 1000000	Driver ID, zero means ID unknown
id_version_char	1.45.03	Text	Up to 8 characters	SW Version, zero means “version unknown"
id_runtime_s	2598000	unsigned Interger	32 bit	Time since last power-on of telematics system [s] , zero means runtime unknown
id_status_char	run	Text	Up to 50 characters	Status of the system

Table 11: system_id: id of the mobile system.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
can_speed_mps	25.36	signed Real	0.00 … 100.00	Master vehicle speed from CAN-bus [m/s]
can_acc_long_mps2	2.45	signed Real	-20.00 … 20.00	Longitudinal acceleration from the CAN-bus [m/s^2]
can_acc_lat_mps2	0.78	signed Real	-20.00 … 20.00	Lateral acceleration from the CAN-bus [m/s^2]
can_acc_long_mean_mps2	2.45	signed Real	-20.00 … 20.00	Medium of longitudinal acceleration from the CAN-bus [m/s^2]
can_acc_lat_mean_mps2	0.78	signed Real	-20.00 … 20.00	Medium of lateral acceleration from the CAN-bus [m/s^2]
can_acc_long_var_m2ps4	2.45	unsigned Real	0.00 … 20.00	Variance of longitudinal acceleration from the CAN-bus [m^2/s^4]
can_acc_lat_var_m2ps4	0.78	unsigned Real	0.00 … 20.00	Variance of lateral acceleration from the CAN-bus [m^2/s^4]
can_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 12: can_1: received can data.

Field	Example	Type	Range	Comment
ts_ms	1378115114000	unsigned Integer	64 bit	Timestamp [ms]
imu_speed_mps	25.36	signed Real	0.00 … 100.00	Speed calculated from IMU [m/s]
imu_acc_long_mps2	2.45	signed Real	-20.00 … 20.00	Longitudinal acceleration from IMU [m/s^2]
imu_acc_lat_mps2	0.78	signed	-20.00 …	Lateral acceleration

		Real	20.00	from IMU [m/s^2]
imu_acc_long_mean_mps2	2.45	signed Real	-20.00 … 20.00	Medium of longitudinal acceleration from IMU [m/s^2]
imu_acc_lat_mean_mps2	0.78	signed Real	-20.00 … 20.00	Medium of lateral acceleration from IMU [m/s^2]
imu_acc_long_var_m2ps4	2.45	unsigned Real	0.00 … 20.00	Variance of longitudinal acceleration from IMU [m^2/s^4]
imu_acc_lat_var_m2ps4	0.78	unsigned Real	0.00 … 20.00	Variance of lateral acceleration from IMU [m^2/s^4]
imu_valid_b	1	Boolean	0, 1	1.. data valid 0.. data not valid

Table 13: imu_1: received data from Inertial Measurement Unit.

2.2 Interface specification

Following the approach introduced for other data sources, in particular the ones that are in charge to collect data from the Traffic Control Centre of Bolzano, the strategy has been to identify the simplest way to cover the last path of the communication chain between mobile system and Supervisor Centre (i.e. vehicle front-end <-> vehicle data source). Since the amount of data could be in the future considerable big, assuming not only the wide set of parameters covered but also the update rate of this communication and moreover the number of probes covered by the system, there has been the need to choose a format which was the most essential as possible. For this reason, the choice has fallen on **JSON** (*JavaScript Object Notation*), which is a rather new data interchange format which is famous for being very lightweight, and being readable as text also easy for humans to analyze and read it. An example of full JSON message is reported below.


```
{
   "carData":[
      {"vehicle_id":"sim1", "values":[
          {"ts_ms":1372659090000,
           "no2_1_ppb":82,
           "no2_1_runtime_s":8569,
           "no2_1_valid_b":1,
           "no2_2_ppb":84,
           "no2_2_runtime_s":8569,
           "no2_2_valid_b":1,
           "o3_1_ppb":75,
           "o3_1_runtime_s":8569,
           "o3_1_valid_b": 1,
           "co_1_ppm":75,
           "co_1_runtime_s":8569,
           "co_1_valid_b":1,
           "res_1_a":84,
           "res_1_runtime_s":8569,
           "res_1_valid_b":1,
           "res_2_a":8569,
           "res_2_runtime_s":8569,
```

```
            "res_2_valid_b":1,
            "temp_1_c":23.2,
            "temp_1_valid_b":1,
            "rh_1_pct":47.3,
            "rh_1_valid_b":1,
            "af_1_sccm":150,
            "af_1_valid_b":1,
            "gps_1_long_deg":16.244373,
            "gps_1_lat_deg":48.343838,
            "gps_1_alt_m":125.3,
            "gps_1_speed_mps":36.2,
            "gps_1_hdg_deg":184.70,
            "gps_1_pdop_nr":1.8,
            "id_vehicle_nr":50,
            "id_system_nr":4,
            "id_driver_nr":69,
            "id_version_char":"1.45.03",
            "id_runtime_s":2598000,
            "id_status_char":"run",
            "can_speed_mps":25.36,
            "can_acc_long_mps2":2.45,
            "can_acc_lat_mps2":0.78,
            "can_acc_long_mean_mps2":2.45,
            "can_acc_lat_mean_mps2":0.78,
            "can_acc_long_var_m2ps4":2.45,
            "can_acc_lat_var_m2ps4":0.78,
            "imu_speed_mps":25.36,
            "imu_acc_long_mps2":2.45,
            "imu_acc_lat_mps2":0.78,
            "imu_acc_long_mean_mps2":2.45,
            "imu_acc_lat_mean_mps2":0.78,
            "imu_acc_long_var_m2ps4":2.45,
            "imu_acc_lat_var_m2ps4":0.78,
            "imu_valid_b":1
            }]
      }]
}
```


The data is transmitted regularly (e.g. once a minute) through a push method from the vehicle front-end to the vehicle data source. The data source must be therefore to be always alive in order to properly receive all data transmission, and check if the JSON message is formatted as expected and contains data which is not already stored at the data center layer.

The data can be transmitted through two methods:

- as a file containing valid JSON, by executing a HTTP post, e.g. to http://ds.integreen-life.bz.it/vehicleds/file;
- as a plain JSON, by posting the data through a HTTP post, e.g. to http://ds.integreen-life.bz.it/vehicleds/json.

AIT and TIS have also agreed to introduce a method that allows the vehicle front-end to know the latest record (in time, i.e. characterized by the most recent timestamp) of a specific vehicle and thus to synchronize the new data transmissions in case of a connection failure. This method could be simply called through a HTTP GET request pointing at a specific Domain Name Server (DNS) with the id of the mobile of interest. The request could be something like:

```
http://ds.integreen-life.bz.it/vehicleds/get-last-record-
timestamp?vehicle=2
```

2.3 Real-time public transportation AVM adapter

As already stated, project partners found a significant added value the possibility to investigate and implement a method to collect the real-time information coming from the AVM system of SASA, the public transport operator covering, among the others, the urban (and inter-urban) service in Bolzano. It is worth to underline that the decision of SASA to introduce with own funding an AVM system in Bolzano for the remote monitoring of the operations of its buses and to let a relevant set of this data available for third parties as open data in standard format, following the German specifications **VDV-453** [7] were significantly influenced by the activities carried out in INTEGREEN, and on top of the strict cooperation that has been possible to build with this organization within the stakeholder involvement activities of Task 6.3.

A specific client capable to properly receive, interpret and store this real-time data stream has been studied and later developed by an external company involved in the project, R3GIS s.r.l,, which has been involved moreover in the design and implementation layer of the Supervisor Centre (presented in more details in the following chapter). Moreover, this client will provide an open API to third parties willing to take advantage of this data in an end-user application. The API will be characterized by a couple of services:

- a service based on **JSON** formatted messages for the initialization of the lines characterizing the public transportation network;
- a **REST** service for the gathering of the real-time positions of the buses and other relevant points of interest, in particular the active bus stops.

Most of the requests to these services accept the parameter **jsonp** so that the output can be in this more compact format, which could be particularly useful for HTML5 applications. The specifications of these services are presented in the tables below.

Request	Description	Example
SASABus.init(targetDivId)	It initializes the map on a specified div. targetDivId is a string containing the id of the div (e.g. “map”).	-
SASABus.showLines(lines)	It visualizes the specified lines. lines is an array of strings, and each of them is an identifier of the line, defined as follows: LINE_NUMBER:VARIATION (e.g. “211:1”)	SASABus.showLines(['211:1', '211:2', '211:3', '201:1']);

SASABus.getLines(success, failure, scope)	It loads all active lines. success is a callback function that is called when data are provided by the server, and the array of lines is passed to the first parameter. failure is a callback function that is called if the response of this request is false /void/null or it has failed. It receives three parameters: xhr, status, error. scope is the scope of the requests of the functions success and failure.	SASABus.getLines(function(lines) { console.log(lines); })
SASABus.getAllLines(success, failure, scope)	It's the same as the previous function. Each line here can have the following fields: (i) li_nr: line; (ii) str_li_var: line variation; (iii) lidname: line name; (iv) li_ri_nr: direction (1: outbound; 2: return); (v) li_r, li_g, li_b: RGB colour	-

Table 14: Real-time buses adapter – JSON service for the retrieval of the public transportation lines information.

Request	**Description**	**Output**
/positions[?jsonp=<nome_funzione>][&lines=<lista linee>]	Requests the positions of the buses	GeoJSON
/{frt_fid}/stops[?jsonp=<nome_funzione>]	Requests the bus stops for the ride identified by frt_fid	GeoJSON
/stops[?jsonp=<nome_funzione>][&lines=<lista linee>]	Requests the list of bus stops	GeoJSON
/{stop}/buses[?jsonp=<nome_funzione>]	Requests the list of buses that will pass through the bus stop "stop"	JSON
/lines[?jsonp=<nome_funzione>]	Requests all active lines	JSON
/lines/all[?jsonp=<nome_funzione>]	Requests all lines (including non active ones)	JSON
/ogc/wms	Wrapper for OGC WMS requests	PNG

Table 15: Real-time buses adapter – REST service for the retrieval of the real-time information.

3 Web interfaces design

The last layer of the Environmental Supervisor Centre is the **front-end layer**. In the original perspective of the project, the idea was to create a **set of simple web interfaces** capable to give an access to the elaborated data and information in the database ("data management unit") that could be used in order to provide information to the local travelers through a demonstrative application, e.g. running on a smartphone and/or any device having at disposal a connection to the Internet. During the requirements' analysis, this perspective was widely extended in the direction of creating a more complex, that could put the premises for the introduction of a virtual market of applications and services destined to end-users through a variety of different business models.

This front-end layer **software platform oriented to service providers** extension, which goes far beyond the initial boundaries of INTEGREEN, was mainly studied and investigated within a complementary **ERDF** project coordinated by the Municipality of Bolzano in cooperation with TIS called "**Bolzano Traffic**". The idea of this project was exactly to permit a multiplication of the channels publishing the RTTI collected through the monitoring system created through INTEGREEN, in order to further increase the impact on the local target audience. This approach is currently the novel trend in *real-time travel and traffic information* (RTTI) distribution, as confirmed by the detailed studies carried out in this project; in this way the data and content provider (in our case the Municipality of Bolzano with all experimental installations on the field) will have to concentrate only on the maintenance of the data delivery services, ensuring that data are always fresh and consistent. **The distribution of information to the end-users is then a responsibility which is left to external service providers**.

In the mobility and traffic sector, providing the right information at the right time to all different users is a very challenging task. Private travelers and commercial operators may present a variety of different needs and habits, which can change also dynamically on the several "profiles" that e.g. a person has to assume (e.g. the need for commercial itineraries could different very significantly to trips moved by leisure reasons, maybe together with the family or friends). In South Tyrol, a very important challenge is related to **occasional tourists**, which are typically the source of heavy congestion phenomena at the entry of the main urban areas of the Province of Bolzano. It is therefore clear how this approach can properly face all this complexity; by properly establishing liaisons with service providers targeting specific end-users communities, one can efficiently reach and cover all this heterogeneous environment. In all this approach, a fundamental role is played by business models based on **open data**: i.e., data will have to be available through standard interfaces to all parties, and win-win models must be defined between content and service providers in order to make possible that the overall traffic distribution chain can be not only economically viable, but can also generate revenues.

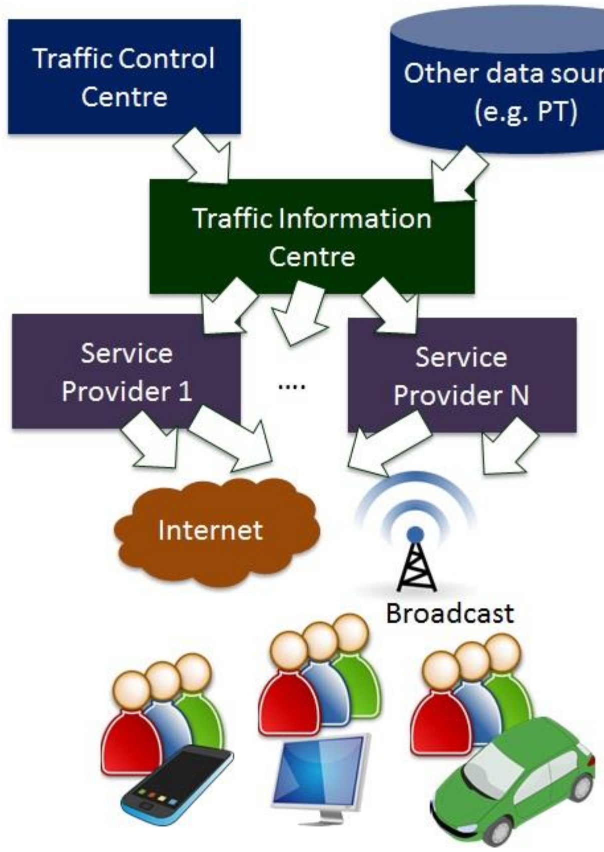

Figure 4: The high-level approach of the front-end layer of the INTEGREEN Supervisor Centre.

Within the project Bolzano Traffic, most of the work has been in the evaluation of the **reference ITS standards** (e.g. DATEX, TPEG, SIRI) that are actually available at an international level for the publication or exchange of traffic information, including the public transportation sector. In particular, the harmonization work carried out in a series of well-known projects funded by the EC through the FP6 and CIP-ICT-PSP programs such as **eMOTION**, **In-Time** and **Co-Cities** has been initially investigated and used as reference for the initial definition of the data model and for the selection and dimensioning of the components needed to build this software platform. In INTEGREEN the design activities of this layer have been specifically two:

- the **extension of the data types** evaluated initially in Bolzano Traffic (which has focused mainly on parking and public transportation information), which need now to include traffic and air pollution measured and elaborated information as well;
- the **definition of a reference GIS stack** able to properly manage the spatial dimension of the managed data.

These design activities have been carried out in partnership with the external subcontractor R3GIS, which has also covered the responsibility to design and implement the interface of the real-time data provided by urban public transport operator SASA, as presented in the previous chapter. The reason for choosing this company has been mainly because it has been involved with TIS in a project co-funded by the program **Interreg Italy-Switzerland** called “**FreeGIS.net**” which has defined both a reference stack and toolkit for the standard publication of geo-information through the use of open data formats (e.g. following the guidelines of the INSPIRE directive and the specifications of OGC) and free and open source software components. In the GIS sector, many tools are today available for the management of spatial data, with different purposes. The goal of this project has been, among others, to put all these different tools together both from a logical point of view, identifying e.g. roles, functions and interfaces, but at a practical level providing a reference toolkit that one can easily use. Of course, what is more important apart of the technological selection of the GIS components is the overall architecture, which well explains the data processing flow and above well takes in proper consideration the proper integration of the **standard OGC interfaces** (e.g. WMS, WFS etc.) [8]- [9]- [10].

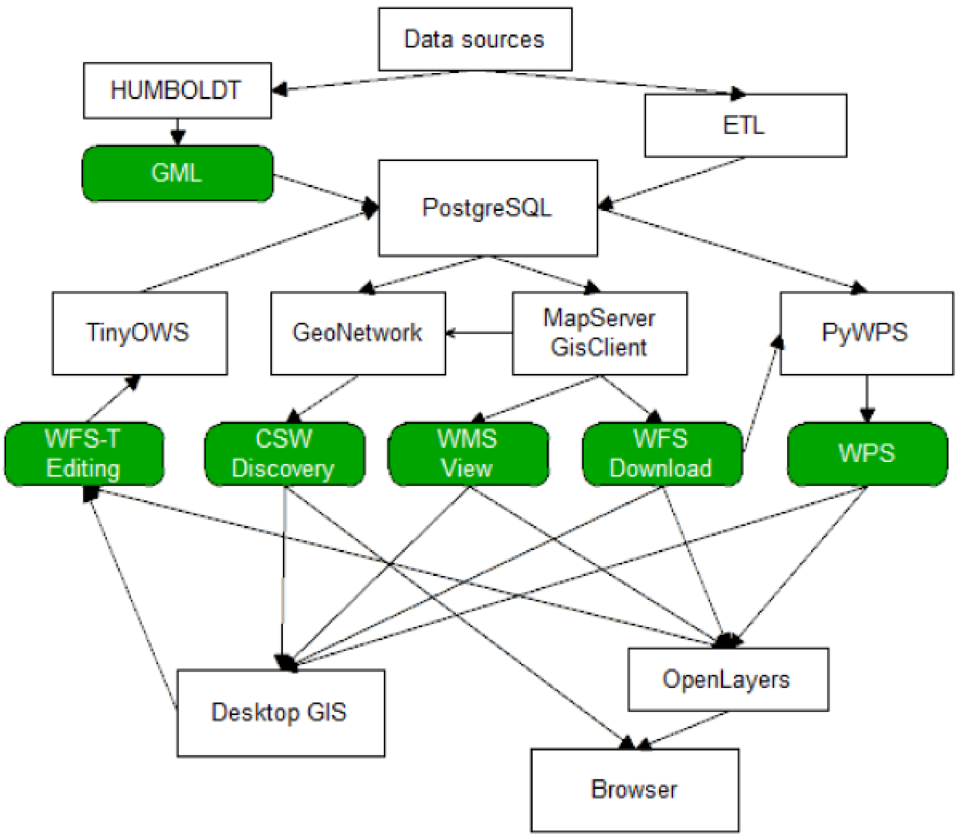

Figure 5: The reference implementation of FreeGIS.net stack.

The purpose of this design work, graphically illustrated in Figure 6, has been therefore to understand **how to match the data publication specifications that must be followed in an ITS environment** (which have been very deeply evaluated in the aforementioned EU projects), **and the ones that are applied in a GIS environment**. The key is to design a data publication environment which is compatible with the open standard specifications defined in both domains.

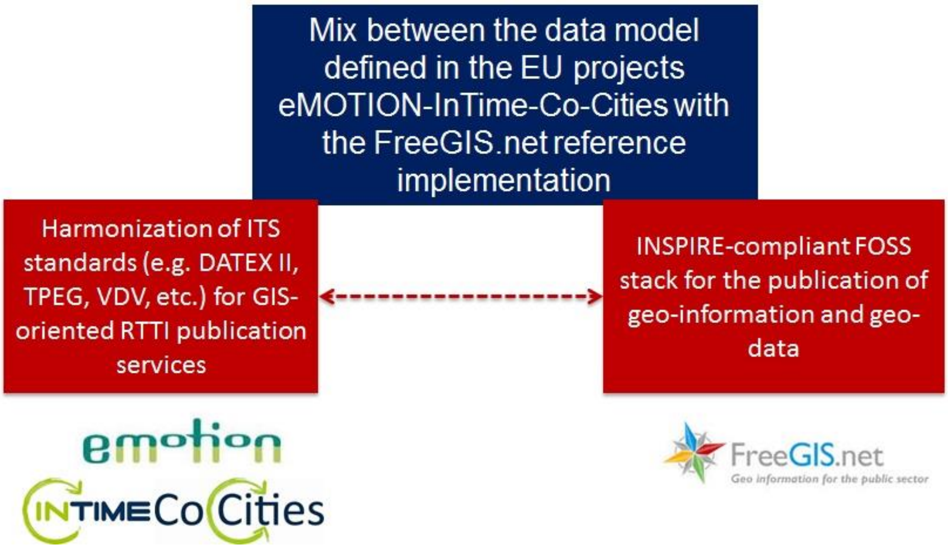

Figure 6: The cross-domain harmonization work carried for the front-end layer of the INTEGREEN Supervisor Centre.

3.1 Front-End layer architecture general design

A set of interfaces that 3rd parties can use in order to access measured and elaborated data stored at the Supervisor Centre has been specifically designed, and are graphically presented in Figure 7. The interfaces that have been selected are the following:

- an interface specifically destined to **other traffic control centers** (and eventually other service providers) through a **DATEX II** based 3rd parties front-end component, following the technological analysis presented in D.3.1.1;

- an interface for **generic service providers** able to offer **OGC- compliant geo-spatial web services**. This will actually be the result of the aforementioned harmonized work since a GIS component (which will be **GeoServer**) will publish the measured and elaborated information stored in the database, which has been specifically designed in order to simplify as much as possible this mapping operation. Based on the functional requirements defined in D.2.1.1, this GIS component can be seen as a natural extension of the data center dispatcher, which has the role to centrally control the external requests coming from the front-ends to the core engine of the platform.

- a variety of **general-purpose web-services**, implemented by several front-ends through different message formats (e.g. XML-RPC, REST, JSON, or others). Through one of this front-end it is possible to introduce a special web-based application to be destined for **traffic operators**.. Thanks to this application, these special users, located at the Traffic Control Centre, will have the possibility to visualize the whole set of data and information collected and computed by the Supervisor Centre, and directly interact with the control systems of the VMS network and the traffic lights that are already in function.

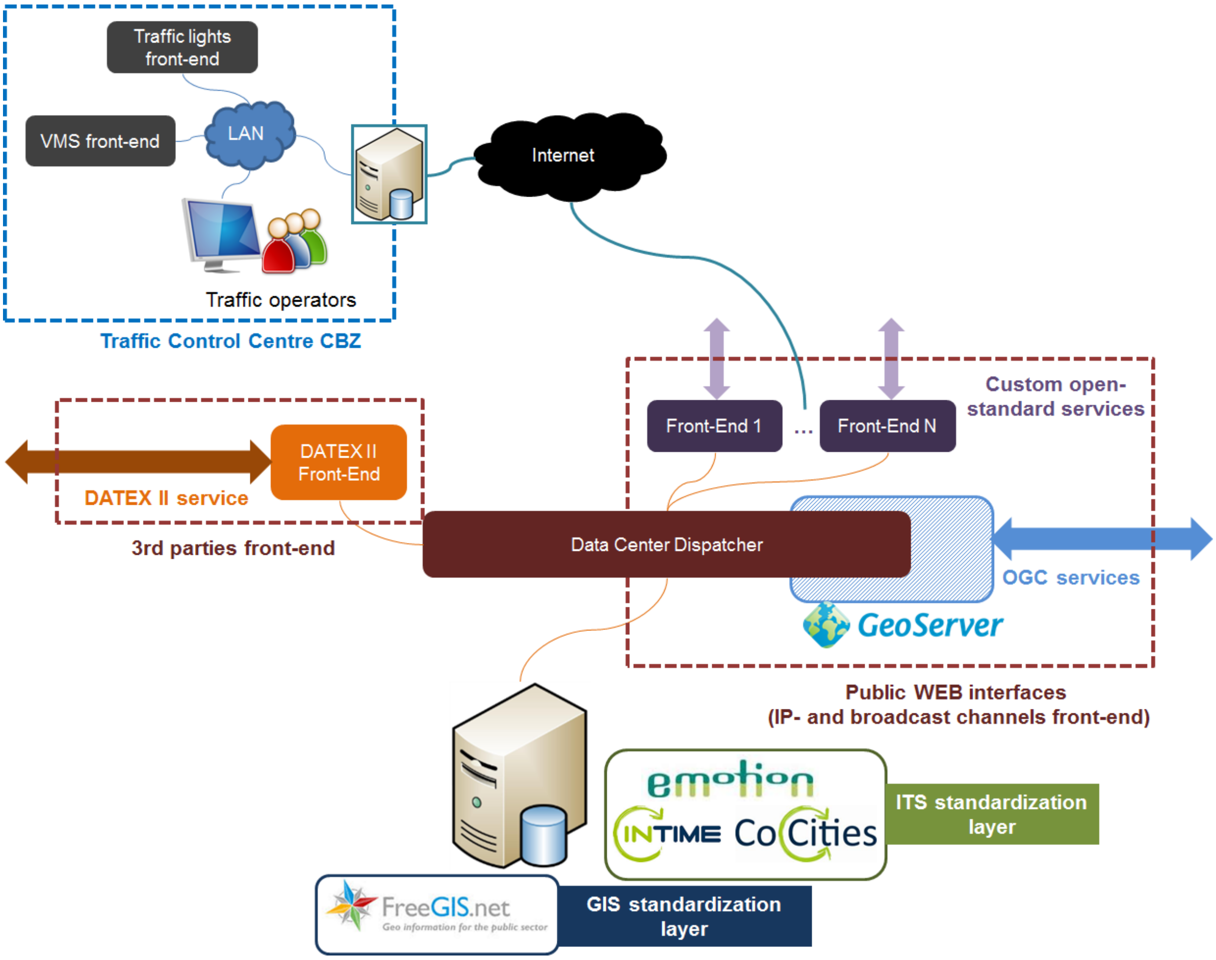

Figure 7: The high-level design of the front-end layer of the INTEGREEN Supervisor Centre.

A set of **demonstrative / commercial applications** will be moreover introduced in order to let final users take advantage of the detailed measurements and elaborations performed in the INTEGREEN core engine.

3.2 Public IP and Broadcast Channels Front-End

As already specified, service providers wanting to create a specific RTTI service over the Internet or over a broadcast channel will have the possibility to get the measured and elaborated information provided by INTEGREEN through two different approaches:

- **through open GIS interfaces**, which will be very useful in case of map representation of the current traffic / environmental conditions in Bolzano;
- **through custom but open standard web services**, which could probably be more efficient in case a 3rd party is more interested in getting and further elaborating the data flow.

3.2.1 *In-Time OGC services*

Following the studies and the results provided in FreeGIS.net, the GIS component of the front-end layer is going to offer a set of **WMS / WFS services** through **GeoServer**, which will have exceptionally at disposal a direct connection to the database at the data center layer. GeoServer has been preferred to other similar GIS tools such as **MapServer** mainly because in this application domain it allows a simpler mapping with the database and is written in Java language, which is the reference programming language chosen for the overall INTEGREEN Supervisor Centre.

Once that the GIS architecture has been specified, there is the need to understand how to guarantee the compliancy with the reference ITS standards, and thus avoid the creation of an technological “island” that other service providers will refuse to use. For this reason, an intense analysis of the path carried out during several EU projects has been carried out in order to properly face this challenge. As already mentioned, the three key projects have been the following:

- **eMOTION**, a project co-financed by the *Sixth Framework Programme* (FP6) of the European Commission and executed in the lifetime 2006-2008 with the aim of introducing a theoretical harmonized data model in the ITS domain covering all existing standards in Europe;

- **In-Time**, a project co-financed by the European Competitiveness and Innovation Programme (CIP-ICT PSP) of the European Commission and executed in the lifetime 2009-2012, which tried to put in practice the data model defined in eMOTION and to empirically test the standardized ITS Business-to-Business (B2B) platform called “Commonly Agreed Interface” in several pilot urban areas in Europe. In-Time was actually the project which not only tried for the first time to implement a fully standardized layer for the publication of RTTI information, but also introduced the novel approach with content and service provider agents that is actually the trend in the ITS sector.

- **Co-Cities**, practically the follow-up of the In-Time project (co-financed by the CIP-ICT PSP programme of the European Commission) and still under execution (2011-2014), which not only tried to further disseminate the In-Time concept, but also to functionally extend its logic through cooperative services, i.e. by adding a feedback loop with the travelers.

The data model introduced in eMOTION is the basic reference for all following pilot activities, and has been widely documented in [11]- [12]- [13]. In particular, in [12] a set of harmonized packages has been defined, covering all relevant ITS domains, in particular:

- the package “**network**”, which models the static networks on top of which the RTTI services can be built, i.e. the road network and the public transportation network;

- the package “**location reference**”, which models the variety of methods available at the state of art which are used in order to create an association between an information and the reference spatial element (which can be generically a point, a line or an area);

- the package “**traffic related information**”, which models the modalities with which traffic information should be published. It follows basically the DATEX II approach, with a clear distinction between the publication of measured / elaborated information and traffic related situations;

- the package “**fixed infrastructure**”, which models with different detail levels the points of interests that characterize the networks;

- the package “**public transport**”, which refers to all specific aspects concerning the publication of information concerning the public transportation services;

- the package “**weather**”, which models the modalities with which environmental information must be published;

- the package “**journey planning**”, which defines the reference data model for journey planning services.

The eMOTION project has given the possibility to download for free this data model, which is available at http://srvweb01.softeco.it/eMotion/site/337/default.aspx The data model can be easily analyzed through UML analysis and modelling tools, in particular Enterprise Architects by Sparx Systems that manages EAP files.

In In-Time, the effort has been to understand how to apply in practice this extensive data model. In fact, in most of the cases public authorities have at disposal a very small subset of the entire dataset which is covered by the **eMOTION data model**, with the need to significantly reduce it. This “simplification” operation is well documented in [14]- [15]- [16]; in particular, in [15] a set of **In-Time services** has been defined in order to create this empirical mapping between the eMOTION data model and practical methods to be deployed in EU cities. This set of services has been defined on top of the technological and organizational state-of-art which was observed in the pilot cities of the project, and cover:

- **In-Time Service 1 – Static Road Traffic Information** (which is mainly the mapping of the eMOTION package “network”);

- **In-Time Service 2 – Dynamic Road Traffic Information on higher network** and **In-Time Service 6 – Dynamic Road Traffic Information for secondary road network** (which are a simplification of the eMOTION package “traffic related information”);

- **In-Time Service 3 – Static Parking Information** and **In-Time Service 9 – Dynamic Parking Information** (which are mainly the mapping of a specific sub-package of the eMOTION packages "fixed infrastructure" and "traffic related information", respectively);

- **In-Time Service 4 – Static Public Transport Information** and **In-Time Service 7 Dynamic Public Transport Information** (which are mainly the mapping of the eMOTION package "public transport");

- **In-Time Service 5 – Walking Information**;

- **In-Time Service 8 – Dynamic Public Transport Journey Routing** and **In-Time Service 17 – Comparative Dynamic Multi Modal Journey Planning** (which are mainly the mapping of the eMOTION package "journey planning");

- **In-Time Service 10 – Enhanced Walking Planning;**

- **In-Time Service 11 – Dynamic Cycling Planning;**

- **In-Time Service 12 – Dynamic Freight Traffic Information** (a specific declination of the eMOTION package "traffic related information");

- **In-Time Service 13 – Dynamic POI Information;**

- **In-Time Service 14 – Dynamic Traffic Event Information** (a mapping of the traffic related situation part of the eMOTION package "traffic related information");

- **In-Time Service 15 – Dynamic Weather Information** (i.e. the mapping of the eMOTION package "weather");

- **In-Time Service 16 – Static and Dynamic Flight Information**.

In-Time has provided similarly to eMOTION the reference UML and WSDL for all these services, so one can simply think to use this reference implementation and adapt it on the base of its particular needs.

In INTEGREEN, most of the design work has been in the analysis of this mapping operation, in understanding which In-Time services could be suitable for the purpose of this project, and eventually revise them by cutting less important parts or integrating data model components defined in eMOTION. While most of the services can fit well the needs of INTEGREEN, there is a key innovation which at present was never tested in a Field Operational Test, i.e. the possibility to include measured and elaborated information in the data packages. In fact, until now pilot areas have only experimented the traffic related situation part of the model, i.e. they have just reasoned in terms of "traffic-related events" and not in terms of continuous measurements and elaborations. Being in this case this second part available for such a use, this domain could be added in the city of Bolzano and theoretically lead to much richer and probably effective services to the end-users.

Based on the previous list of In-Time services, in INTEGREEN the focus will cover on:

- **In-Time Service 1 – Static Road Traffic Information**
- **In-Time Service 2 / 6 Dynamic Road Traffic Information**
- **In-Time Service 3 – Static Parking Information** and **In-Time Service 9 – Dynamic Parking Information**
- **In-Time Service 14 – Dynamic Traffic Event Information**
- **In-Time Service 15 – Dynamic Weather Information**

It is worth to underline how the most peculiar service for INTEGREEN is actually the service 15, because of the relevancy of the environmental issues. For completeness sake, in the next pages we have reported the reference UML diagrams of these services given by the In-Time data packages.

INTEGREEN / In-Time Service 1 – Static Road Traffic Information

Through this service, a service provider could get the static road traffic information through a WFS service (in the example, *getStaticRoadData*). Please note that public transportation network information are not covered here.

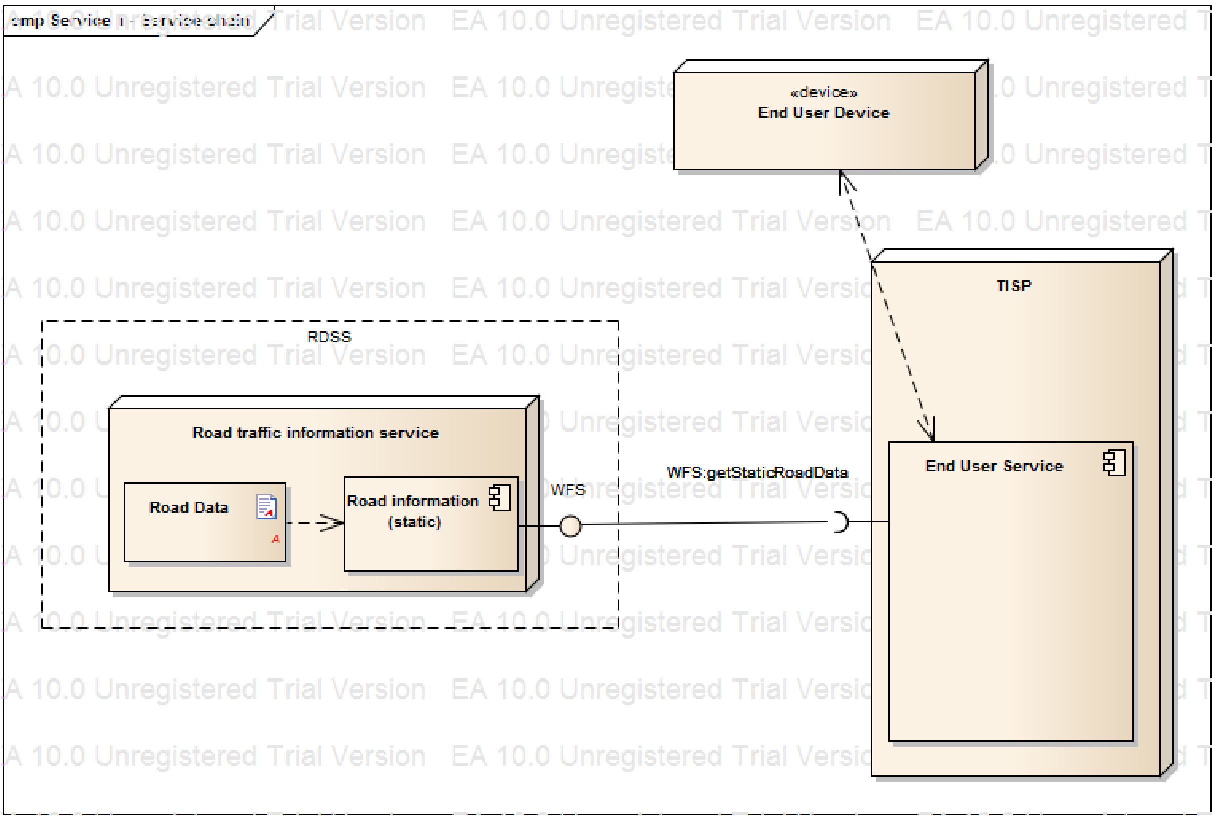

Figure 8: High-level design of WFS static road traffic information service.

Through the detailed information at disposal of the Municipality of Bolzano, it will theoretically possible to give all relevant details of the road network. The ferry link classes will not considered in the INTEGREEN data model.

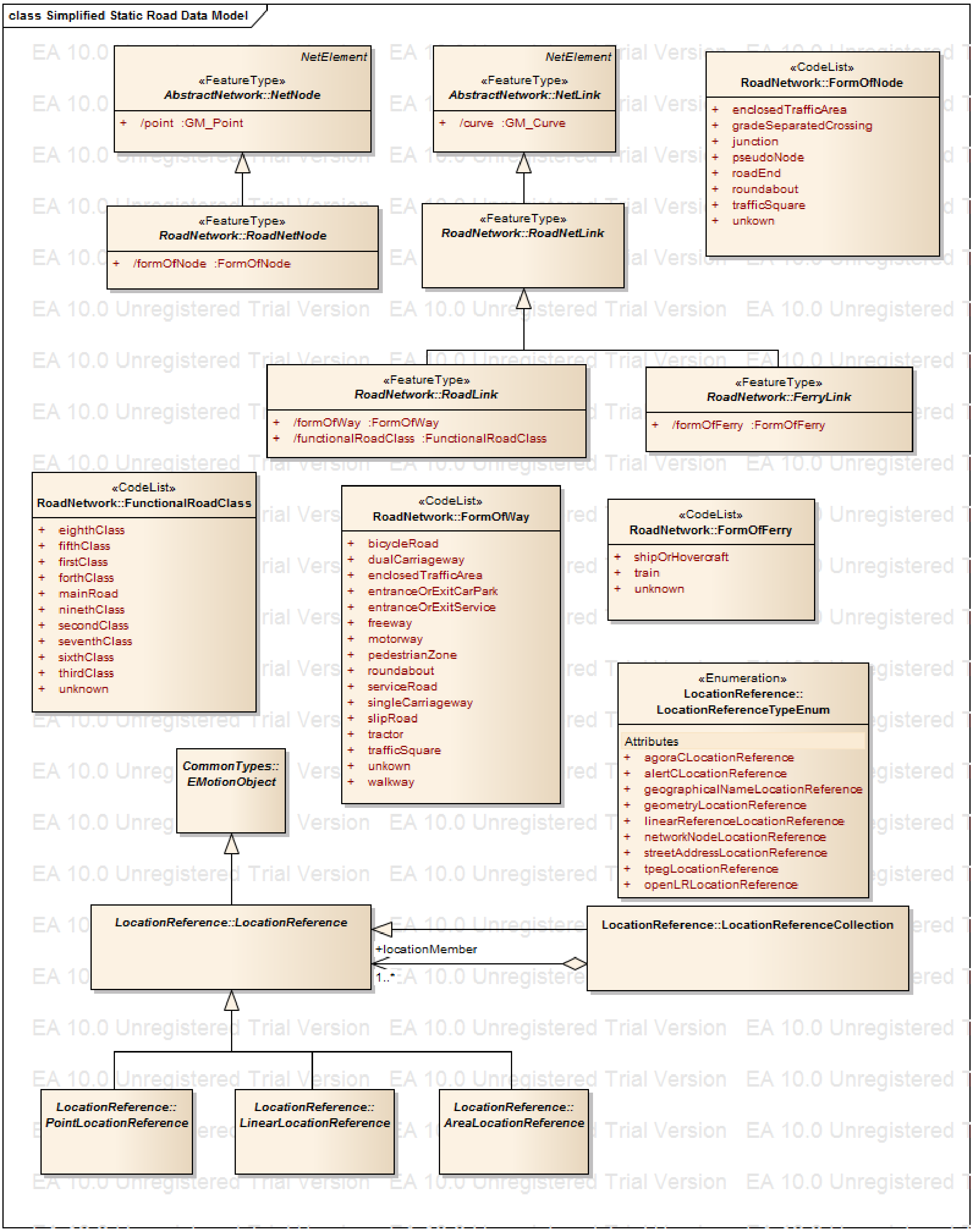

Figure 9: The reference UML model of the static road traffic information service.

INTEGREEN / In-Time Service 2 – Dynamic Road Traffic Information

Through this service, a service provider could get the dynamic road traffic information through a WFS service (in the example, *getDynamicRoadTraffic*).

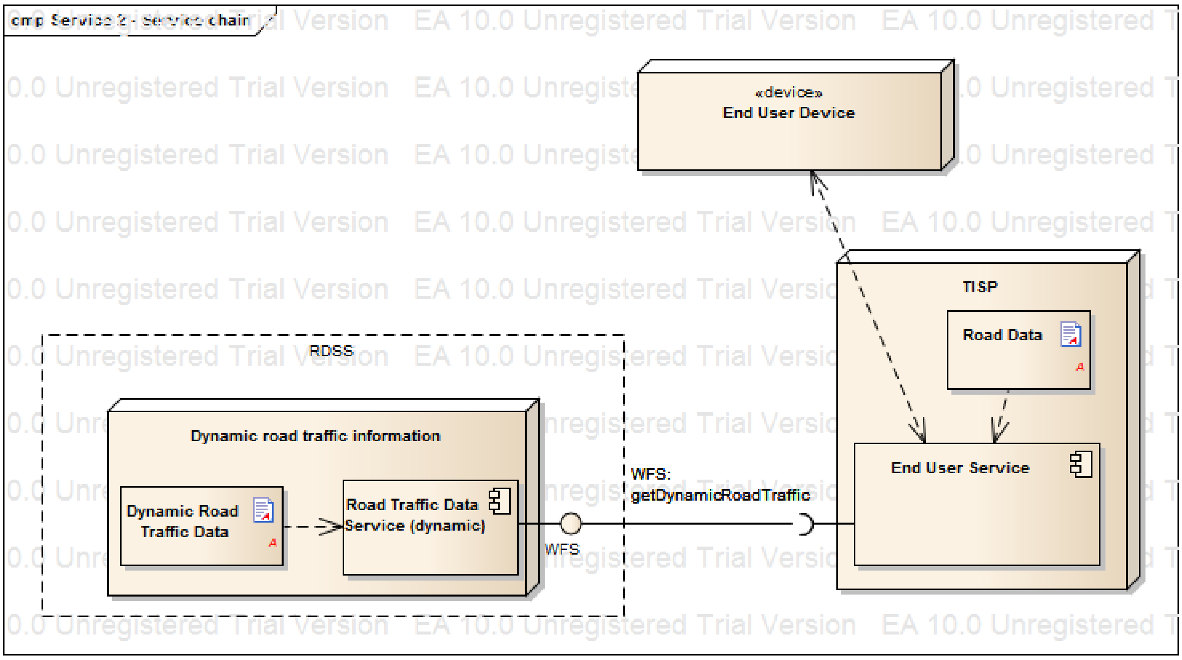

Figure 10: High-level design of WFS dynamic road traffic information service.

The idea here is to populate the service not only with **traffic events** (e.g. programmable or spontaneous) and **operators actions** (e.g. road works) information, as illustrated in Figure 11, but also measured and elaborated data, as specified in the DATEX II standard and indicated in the eMOTION data model as well. This can include among other indications such as travel times, traffic status, traffic values, etc. Some additional UML diagrams directly taken from the latter model has been added and are reported in the following pages. The main class *TrafficRelatedInformationFeature* will be practically enriched by classes *MeasuredDataPublication* and *ElaborationDataPublication*, as illustrated in Figure 12 . It is worth noting how a similar approach could be followed also for the environmental service (later presented), as shown in Figure 14.

It is worth noting how this service is capable to deliver **detour recommendations**, which can include among others network management and traffic control information that could be dynamically altered by traffic operators. This could be an interesting opportunity to immediately communicate the eco-friendly policies suggested by INTEGREEN to local travelers, in particular in direction of cooperative applications.

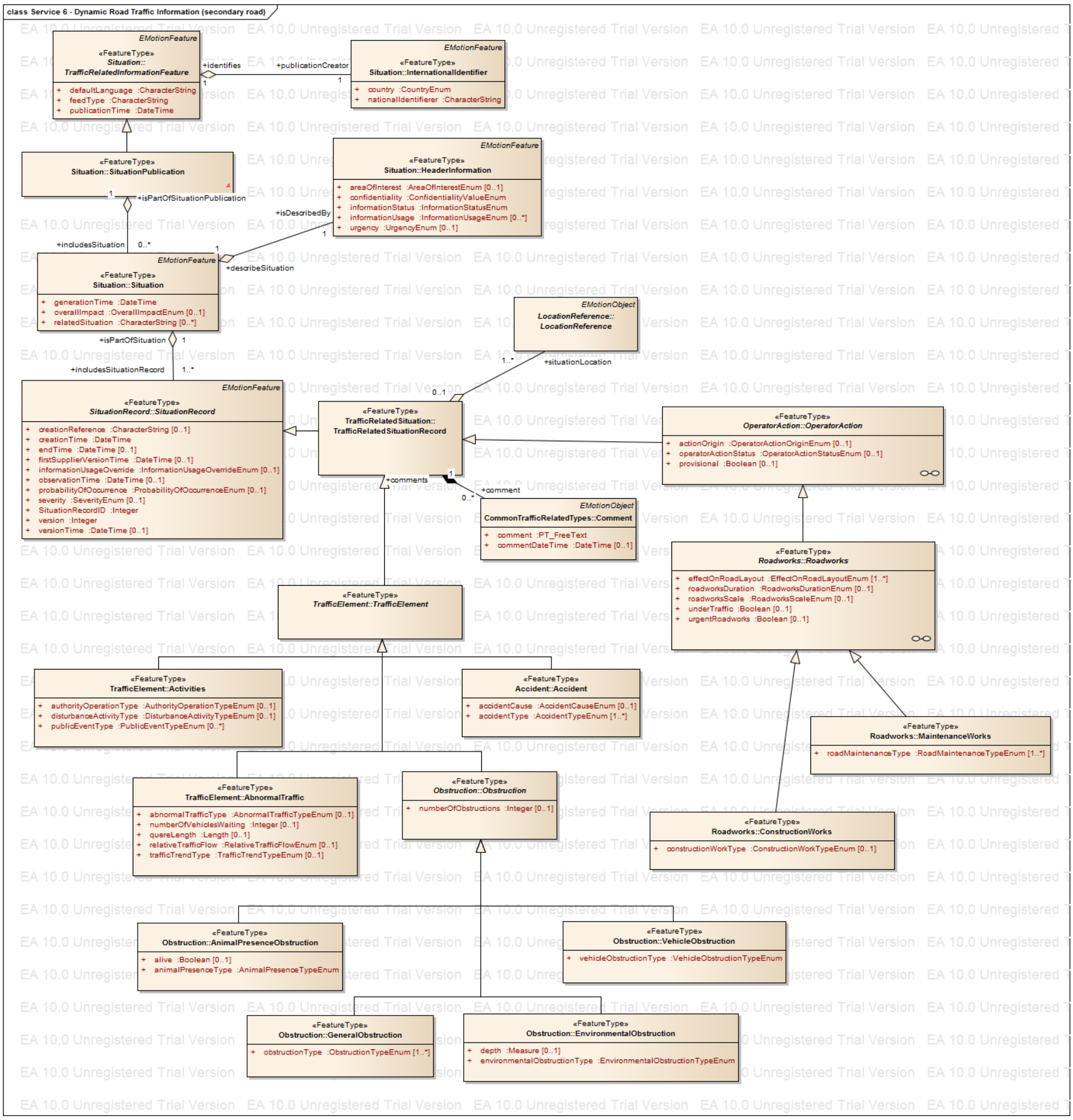

Figure 11: The reference UML model of the dynamic road traffic information service.

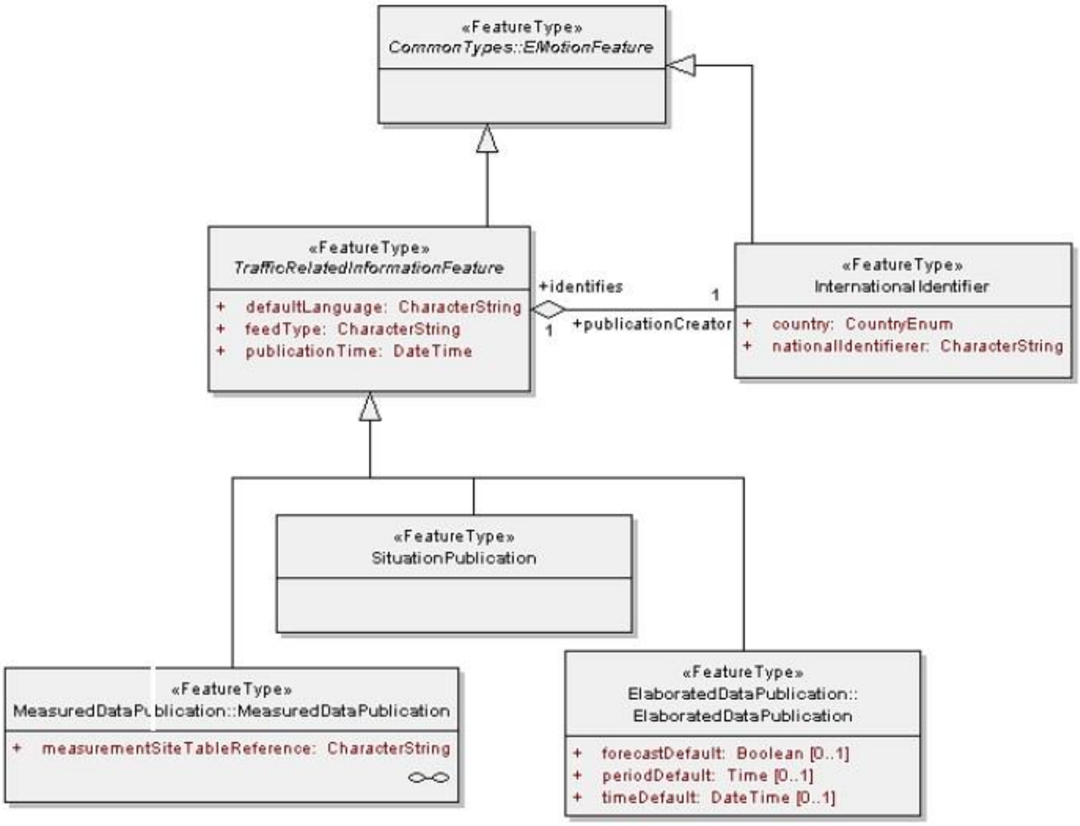

Figure 12: The reference eMOTION UML model for the publication of all traffic related information.

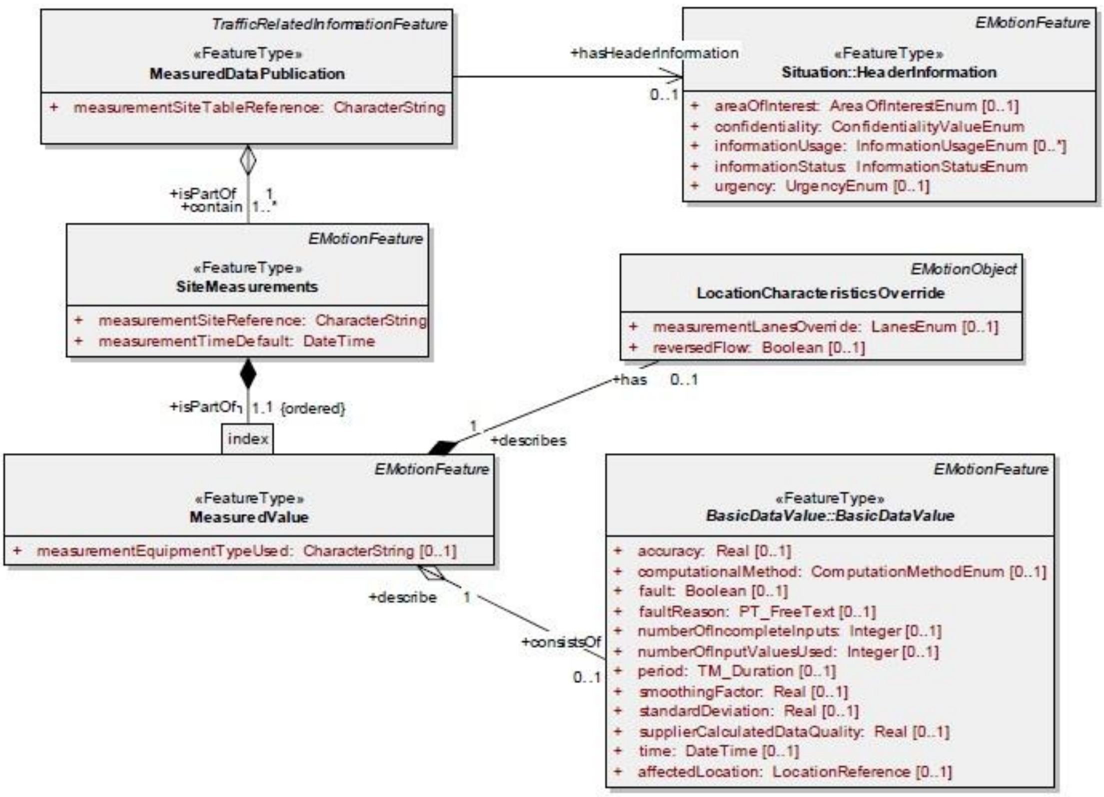

Figure 13: The reference eMOTION UML model for measured data publication.

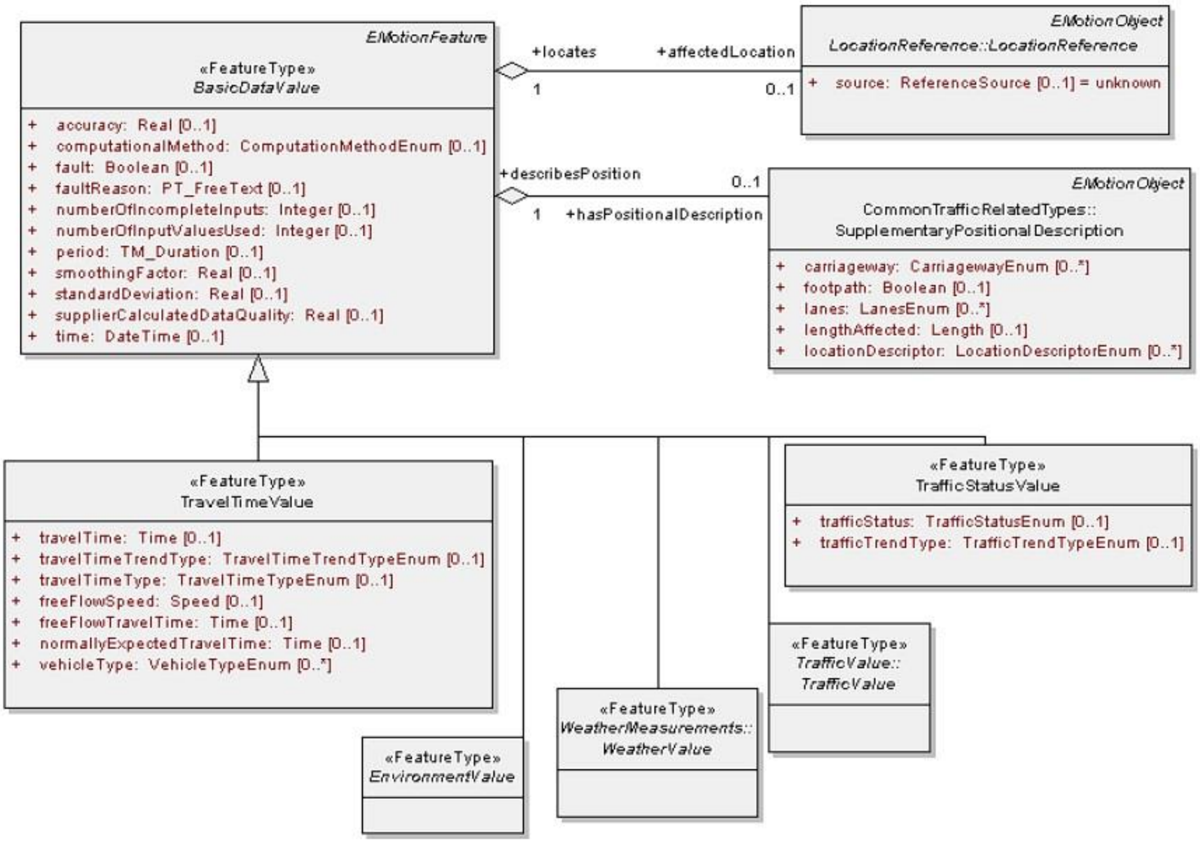

Figure 14: The reference eMOTION UML model detail for traffic measurements.

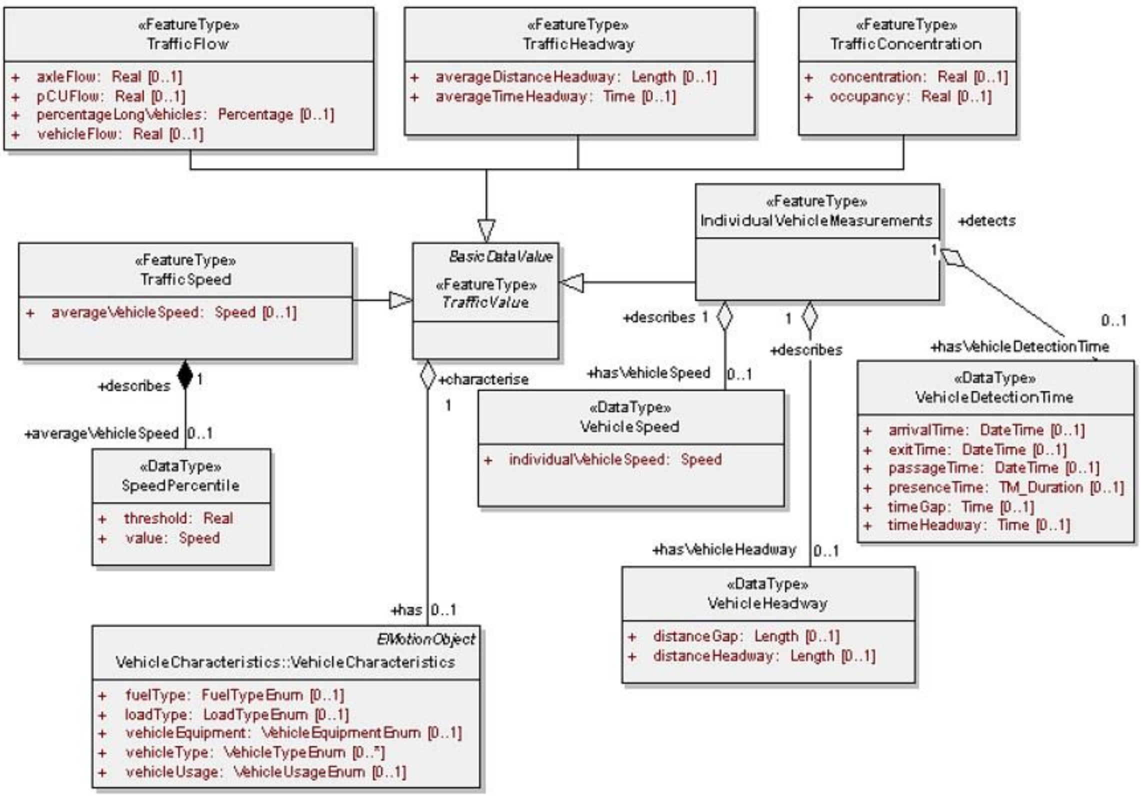

Figure 15: The reference eMOTION UML model detail for traffic measurements (part 2).

INTEGREEN / In-Time Service 3 – Dynamic Parking Information

Through this service, a service provider could get the dynamic parking information through a WFS service (in the example, *getDynamicParkingInformation*). This service manages static information about parking areas as well.

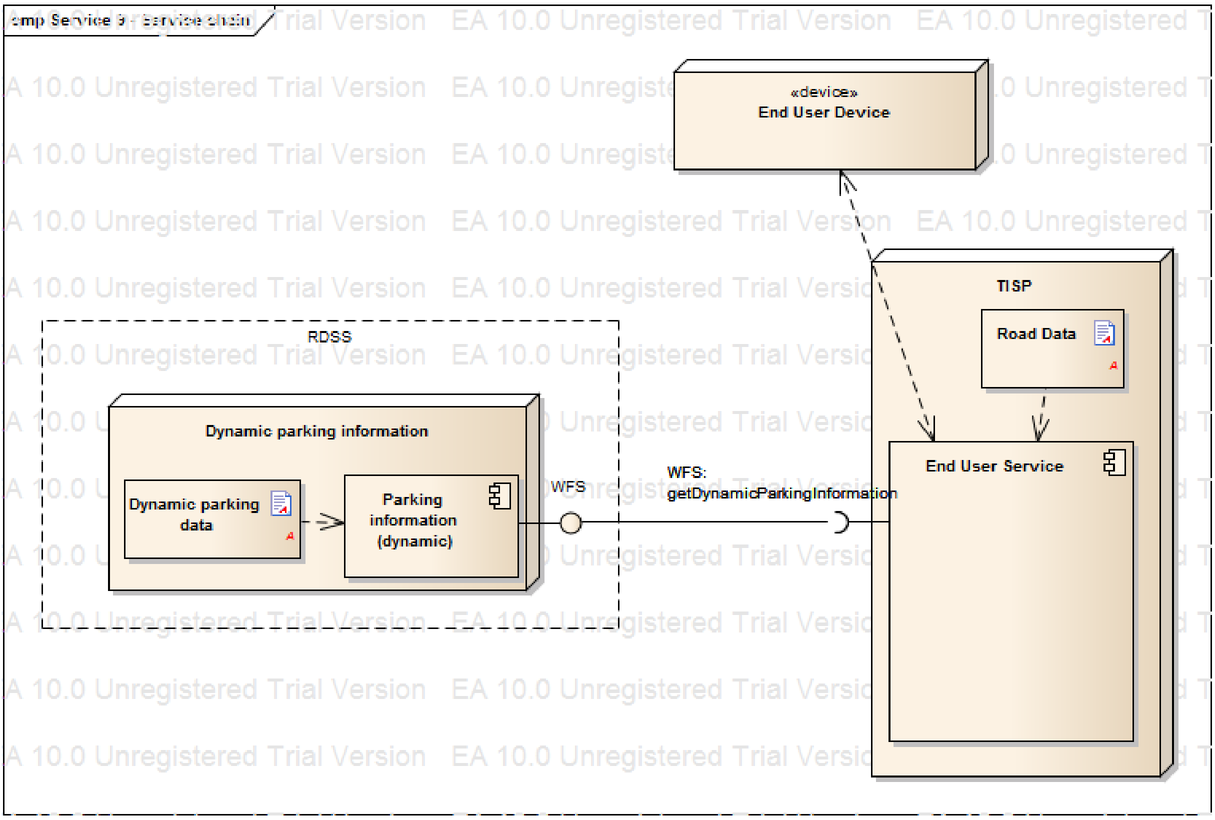

Figure 16: High-level design of WFS dynamic parking information service.

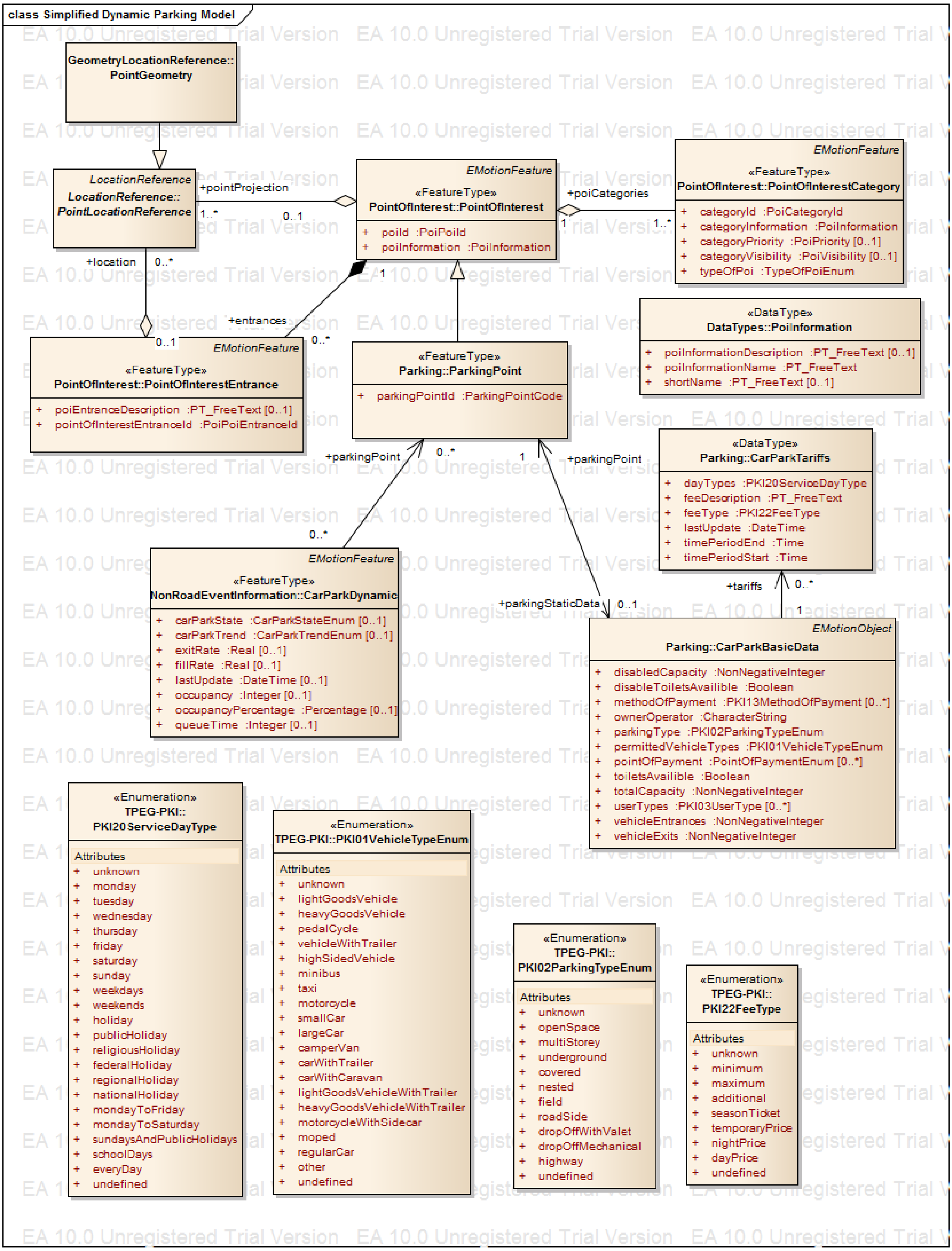

Figure 17: The reference UML model of the dynamic parking information service.

INTEGREEN / In-Time Service 4 – Dynamic Weather Information

Through this service, a service provider could get the dynamic weather information through a WFS service (in the example, *getDynamicWeatherInformation*). This service is designed as as service 2 (i.e. can contain measured and elaborated data as well) but deals with environmental information only.

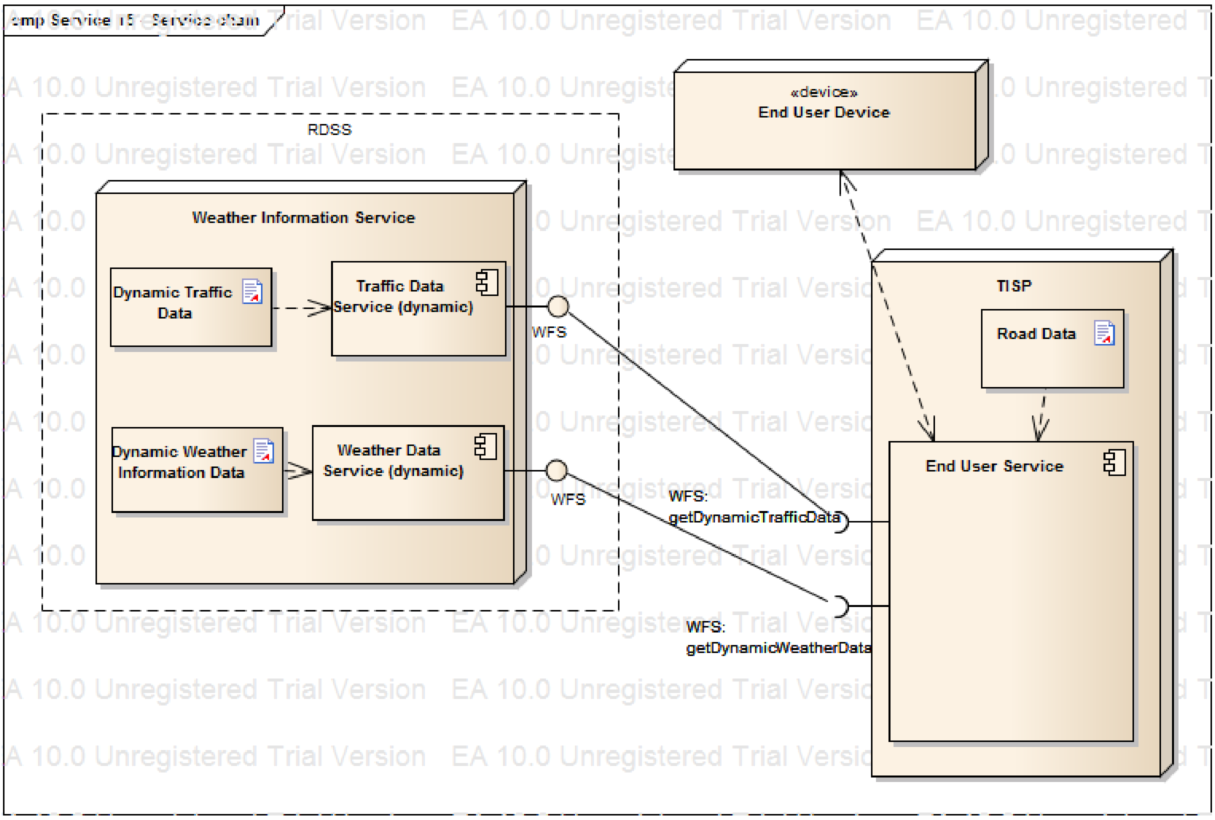

Figure 18: High-level design of WFS dynamic weather information service.

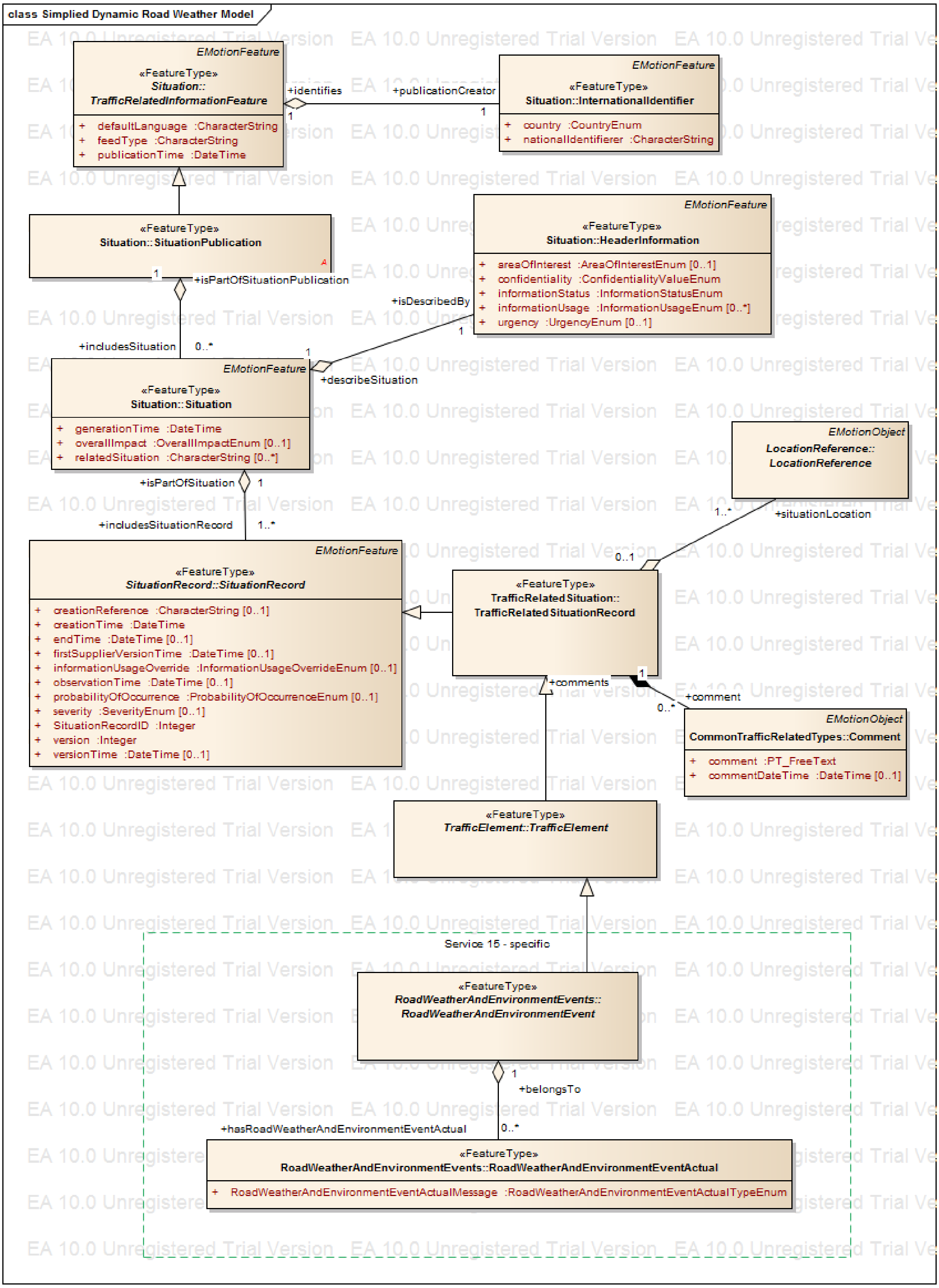

Figure 19: The reference UML model of the dynamic weather information service.

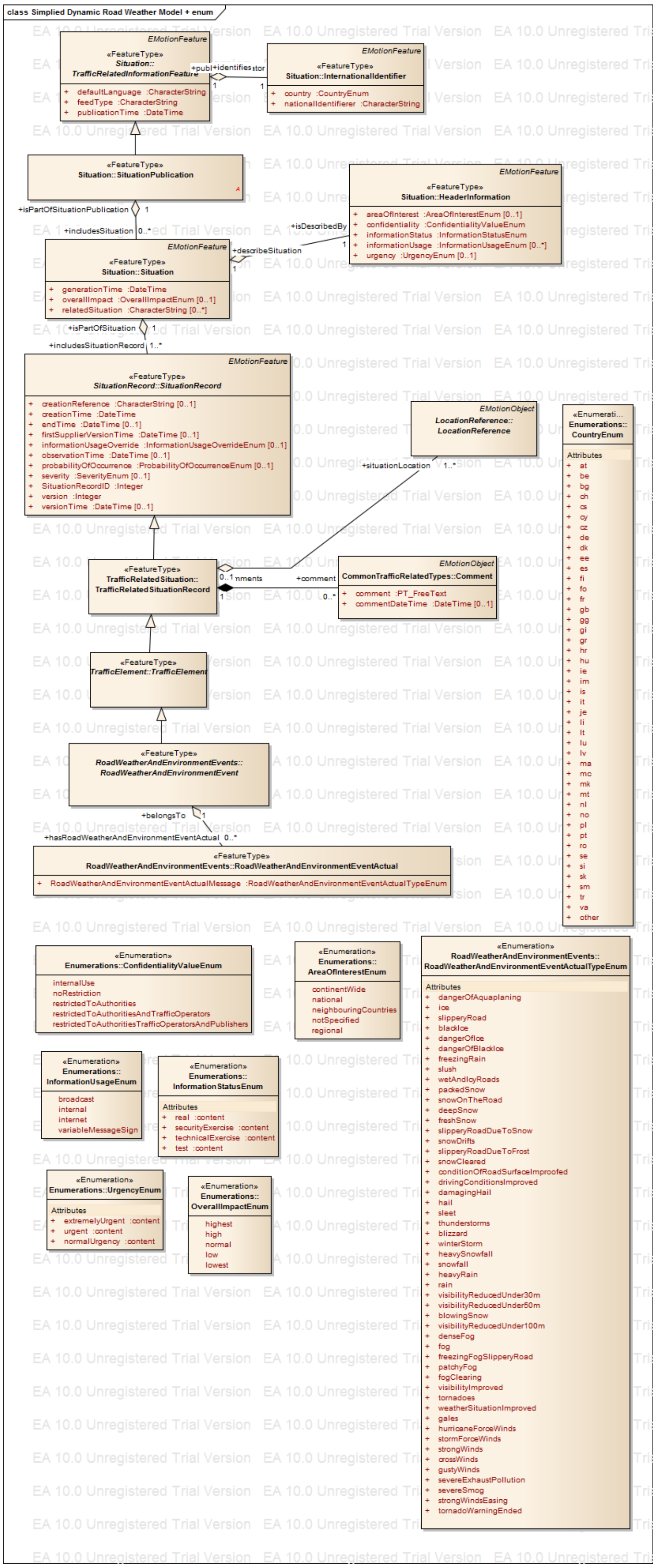

Figure 20: A detailed overview of the UML model of the dynamic weather information service.

3.2.2 *Custom open-standard services*

The reference use case beyond this set of custom open-standard services is the one related to the web interface destined to the traffic operators, which must be capable to handle all measurements and elaborations gathered and stored in the data center layer of the Supervisor Centre. These services will therefore be the dual part of the active data sources, and each them will be in charge of requesting specific data types to the data center collector and offer them through standard web technologies to 3rd parties applications. Two basic profiles must be foreseen, namely:

- **privileged user**, who has the role to access and visualize everything. This will be an exceptional case (i.e. traffic operators), and only a specific application (i.e. BZAnalytics) will have the right to request all measured and elaborated information;

- **non-priviliged user**, who has the role to access only a specific subset of information that the content provider (i.e. the Municipality of Bolzano) will decide to make available to third parties.

The roles of heterogeneous users must be managed at the front-end layer; the front-ends must therefore have the capability to receive all relevant information they are managing from the data center dispatcher. The reference technologies which will be used to implement these services are going to be:

- **XML-RPC** for the <u>communication between data center dispatcher and front-ends</u> (similarly as the communication between the data sources and the data center collector);

- **REST** and **JSON** for the <u>services at disposal for third parties</u>.

3.3 3rd Parties Center Front-End

This front-end will be implemented as a **DATEX II** service. Based on the technological analysis of this technology given in D.3.1.1. and the above selection of the OGC services, the decision is to replicate **services 2, 3 and 4** also through a standard DATEX II service. In this way, the information which is shared to external parties is exactly the same, but simply made available in spatial and non-spatial form. Service 1 is not included here because not relevant for this kind of application scenario. Since there is at present no specific need from other parties, the service will be implemented through a **push** mechanism.

3.4 Variable Message Signs and Traffic Lights Center Front-Ends

Given the design choices described in the previous paragraphs and the high-level integration strategy illustrated in D.3.1.1, the operators will continue to use the actual systems for directly operating with VMSs an traffic lights. For the purposes of this demonstrative project, there won’t be automatic tools interfacing with these existing systems, which could be however implemented on top of the results of the FOT.

3.5 End-user demonstrative applications

As far as the **services for end-users** is concerned, it is worth noting how in the frame of the Bolzano Traffic project an extensive set of applications has been first introduced thanks to the cooperation of several local partners. In the first months of the summer season 2014, the final set of applications will be released and local travelers could start using them in order to improve the efficiency of their travel choices, both in the pre-trip planning as well as while en-route. The applications are the following:

- **SASAbus**, which is an application available both as HTML5 and Android which provides static information about the urban public transportation service offered by SASA as well as the real-time locations of the buses, so that users can immediately check their expected arrival time in correspondence of certain bus stops. The application is going to integrate information provided by the Supervisor Centre such as the real-time occupancy of the parking areas, so that it will be possible for users to dynamically check the opportunity for park & ride trip options;

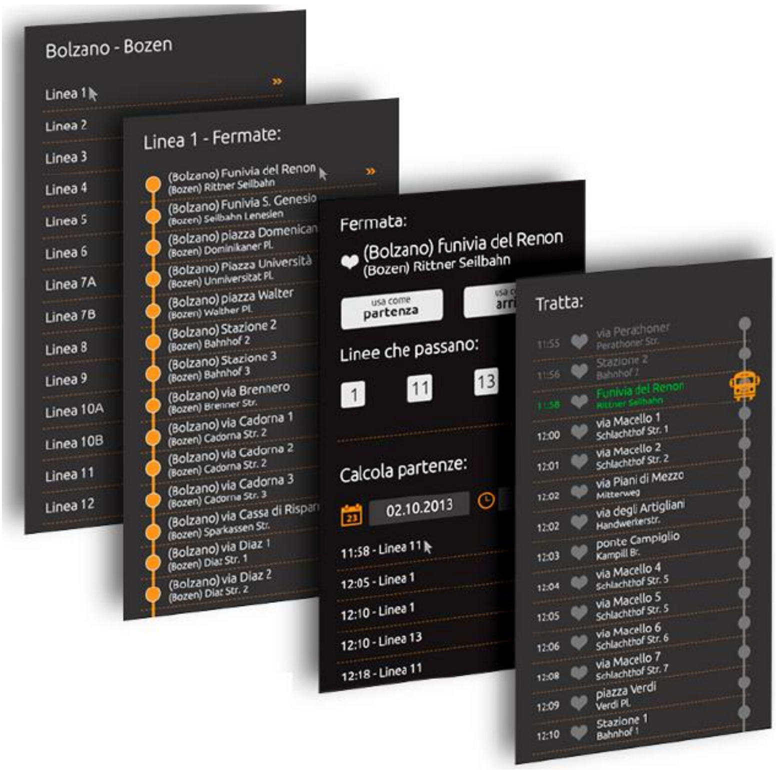

Figure 21: The SASAbus application [11].

- **BZ Parking**, a demonstrative HTML5 application implemented by TIS and other local companies which will present not only the current parking lots availability in the main parking areas of the city but also a short term forecast of how conditions will evolve. This functionality will be particularly useful for tourists evaluating the best trip options for reaching Bolzano from a surrounding valley;

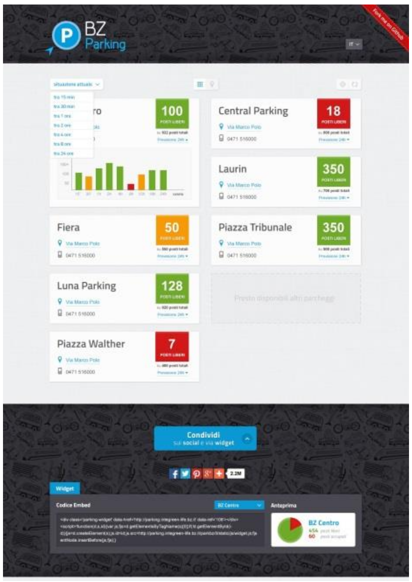

Figure 22: The BZParking application.

- **South Tyrol Suggest**, an Android application which is the result of the research activities carried out in the past years by the **Faculty of Computer Science of the University of Bolzano** which has been further improved thanks to the synergy created in this project. The application is technically a mobile **context-aware recommendation system**, i.e. a system capable to recommend POIs in the local area (e.g. events, tourist attractions, public services, etc.) as a function of the <u>specific preferences of the user</u> and the <u>particular boundary conditions</u> (e.g. meteorological conditions and forecasts, parking availability, but also specific trip details, e.g. alone or with friends, free time or business trip, etc.). STS continuously improves its knowledge about the users' preferences thanks to the evaluation of his / her interactions, and adapts its suggestions on the base of the heterogeneous context information collected through the access to an extended set of (web) services. Moreover, the user can get **<u>"eco-aware" recommendations</u>** about how to plan a trip through a POI, namely the most environmental-friendly trip choice (in terms of transport mode and route choice) is proposed through a very intuitive GUI. Another innovation introduced in the project is related to a **real-time multi-modal routing engine**, which STS as well as other applications such as SASAbus can use as advanced functionality in this trip planning phase;

Figure 23: The South Tyrol Suggest application.

- **Co-Cities**, available as Android and iOS applications, which are the demonstrative applications developed within the EU Co-Cities project and which have been extended in order to include the RTTI provided by the data providers in Bolzano. At present, the application includes again the real-time parking data as well the information given by SASA concerning the urban public transportation service. The advantage of this application is that a traveler could use it in a certain number of European cities (i.e. the pilot areas of the project), without the need / issue to discover the local application(s) providing the RTTI needed for organizing a trip in an unknown area.

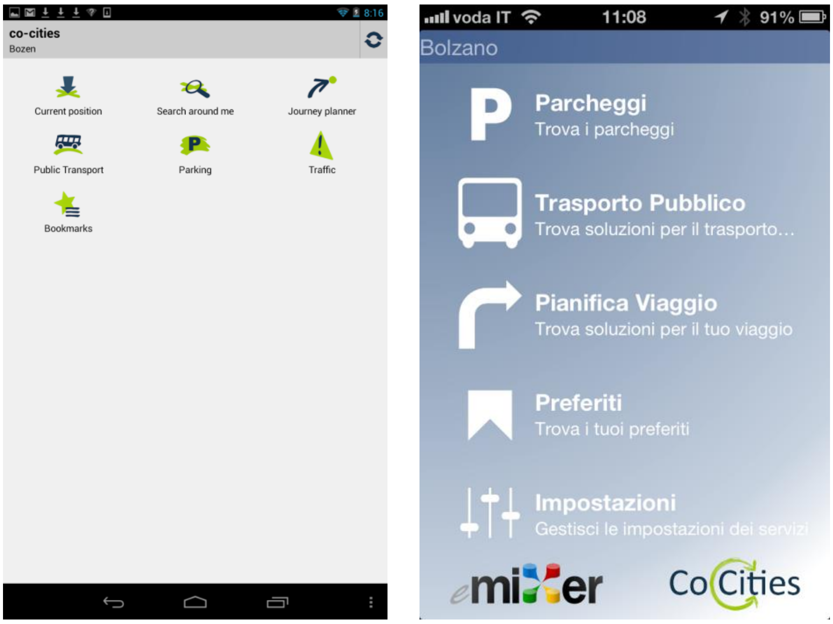

Figure 24: The demonstrative Co-Cities applications.

Other service providers will also multiply the visibility of the local RTTI. For example, through the cooperation with some private companies and local tourist organizations, the idea is to bring this information in the **TVs of the hotel rooms**, or on **next-generation touch screen totems** to be installed in city locations at high visibility from occasional travelers such as tourists. Further cooperation is in plan with media partners, such as **internet portals** and **local broadcaster**, so that they improve the quantity and quality of traffic information they already distribute.

In INTEGREEN, the plan is one side to introduce two **further demonstrative applications**, i.e.:

- one to be probably called “**BZTraffic**”, <u>which is going to present the elaborated traffic conditions as well as some inputs concerning the environmental situation in the city thanks to the advanced monitoring system which will be introduced</u>;
- one to be probably called “**BZBus**”, which is going to present the current positions of the urban bus fleet in Bolzano thanks to the interface with the AVM system of SASA.

It is worth to remember a specific web application destined to traffic operators called “**BZAnalytics**” will be included as well in this applications suite.

On the other side, thanks to the open data approach followed by the project (i.e. the idea to give free and documented access to the data and information that is officially released to the public under certain licenses conditions), the perspective is to further improve and extend the number of information and functionalities of services that are already available in the area. The strategy is indeed more to enable the **creation of a virtual “market” of services** for the end-users developed and maintained by local service providers, rather than having just one exclusive application governed by the Municipality of Bolzano. The more the type, accuracy and amount of data as well the possible elaborations will be available, the more this market could be stimulated, in particular if solid business models will be identified and built.

It is worth noting at this stage that while as far the **traffic data** there is no particular concern, in particular by the Municipality of Bolzano, to release all the available measured data and elaborated information, all this is much more critical if applied to the **air pollution data**, which can produce consequences and outcomes at political level. In order to properly deal with current barriers, the approach would be to provide qualitative indications about the current environmental conditions in the city, which can represent an active incentive to increase the awareness of local users about this concern and therefore additionally foster a positive change of the mobility habits, in particular of occasional travelers.

Conclusions

This report has presented the main design choices covering the vehicle-to-centre front-end and the web interfaces. In particular, the following aspects have been presented:

- the specification of the interface between the vehicle data source at the Supervisor Centre and the vehicle front-end managed by AIT for the **real-time exchange of the data delivered by the mobile stations**;

- the specification of the interface for the **collection of the real-time positions of the urban public transportation service**;

- the high-level and detailed design choices defined for the front-end layer, which can be summarized as follows:

    - the **definition of the OGC services** that will delivery relevant traffic and environmental information in a GIS format, based on a harmonization work that has covered both ITS and GIS domain;

    - the **definition of the custom open standard services**, that will serve primarily the application destined to the traffic operators (**BZAnalytics**) as well as other potential service providers that are not interested in the spatial representation of the information;

    - the **definition of the interface with 3rd parties traffic control and/or management centers**, which will be implemented as a DATEX service;

    - the identification of a set of demonstrative web applications to be offered to local travellers, i.e. **BZTraffic**, **BZBus** and potentially all deployed applications introduced in the scope of the **Bolzano Traffic** project.

Bibliography

[1] INTEGREEN consortium, "D.3.1.1 "Data management unit and environmental stations front-end"," 2014.

[2] INTEGREEN consortium, "D.3.2.1. "On board telematic unit"," 2014.

[3] INTEGREEN consortium, "D.3.2.2 On board traffic and environmental monitoring unit," 2014.

[4] INTEGREEN project, "INTEGREEN proposal: technical forms," 2011.

[5] U.S. Department of Transportation, "Systems Engineering for Intelligent Transportation Systems," 2007.

[6] INTEGREEN consortium, "D.2.2.1: "Mobile system requirements"," 2012.

[7] VDV (Verband Deutscher Verkehrsunternehmen), "VDV 453 "Actual Data Interface - dynamic schedule synchronisation - dynamic passenger information - visualisation - general - information service - Version 2.3.2"," 2013.

[8] F. D'Alesio and P. Hopfgartner, "Free WebGIS Software Analysis," FreeGIS.net project, 2012.

[9] F. D'Alesio, P. Hopfgartner and P. Viskanic, "Review of relevant WebGIS standards," FreeGIS.net project, 2012.

[10] F. D'Alesio, P. Hopfgartner and P. Viskanic, "User stories and Reference Implementation," FreeGIS.net project, 2012.

[11] eMOTION consortium, "D5: "eMOTION System - Analysis of Technical Standards"," 2008.

[12] eMOTION consortium, "D6: "eMOTION System - Technical Specification Appendix 1 eMOTION Data Model"," 2008.

[13] eMOTION consortium, "D6: "eMOTION System Technical Specification Appendix 2: eMOTION Service Interface Definition"," 2008.

[14] In-Time consortium, "D.2.3.1: "Report on In-Time Interfaces and Protocols"," 2010.

[15] In-Time consortium, "D.3.1.2 "UML schemes finalized"," 2010.

[16] In-Time consortium, "D.3.2.1 "Specification Document of In-Time server, interfaces and

protocols"," 2010.

[17] D. Montesin, "SASAbus HTML5 & related Java Free Software - Public transport easy and for everybody," in South Tyrol Free Software Conference 2013, Bolzano, 2013.

[18] Co-Cities consortium, "D3.1 "ITS system specification description and reference platform for validation"," 2012.

[19] Co-Cities consortium, "D3.2 "ITS system interfaces to end users and service delivery chain of the service providers"," 2012.