FP7-SMARTCITIES-2013

STREETLIFE

Steering towards Green and Perceptive Mobility of the Future

WP4 – Mobility Management and Emission Control Panel

D4.1– Requirements analysis, mechanism and technology selection, and control panel specification

Due date: 31.01.2014	**Delivery Date**: 12.02.2014
Editor: Benjamin Dittwald (Fraunhofer)	**Contributors (Co-authors)**: Jens Kohlmorgen (Fraunhoer), Björn Schünemann (Fraunhofer), Birgit Obst (SIEMENS), Veronika Brandstetter (SIEMENS), Philipp Gilka (DLR), Davide Frigeri (CAIRE)
Contributing Partners: Fraunhofer, SIEMENS, DLR, CAIRE	
Dissemination level: Public	**Nature of the Deliverable:** Report
Internal Reviewers: Norbert Reithinger (DFKI), Giuseppe Valetto (FBK)	

Executive Summary:

Deliverable 4.1 describes the requirements and specifications of each task of WP4. Also each task evaluated possible tools to help us to reach the goal of the tasks. Considering the scenarios and use cases of the pilot site and the available components and data of each pilot this deliverable describes an architecture approach to fulfil the requirements of the pilots. It describes the results of the work in WP4 at month 4.

D4.1– Requirements analysis, mechanism and technology selection, and control panel specification

Table of Contents

ABBREVIATIONS

CO	Confidential, only for members of the Consortium (including the Commission Services)
D	Deliverable
DoW	Description of Work
FP7	Seventh Framework Programme
FLOSS	Free / Libre Open Source Software
GUI	Graphical user Interface
IPR	Intellectual Property Rights
MGT	Management
MS	Milestone
OS	Open Source
OSS	Open Source Software
O	Other
P	Prototype
PU	Public
PM	Person Month
R	Report
RTD	Research and Development
WP	Work Package
Y1	Year 1
ITEF	Intelligent Test Evaluation Framework
CIP	City Intelligence Platform
OSM	OpenStreetMap
API	Application Programming Interface
BRA	Business Rule Approach
GTFS	The General Transit Feed Specification

PARTNER NAME CORRESONDING TO ABBREVIATION IN STREETLIFE DOW

Fraunhofer	Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V.
FBK	Fondazione Bruno Kessler
SIEMENS	Siemens AG
DFKI	Deutsches Forschungszentrum für Künstliche Intelligenz GmbH
AALTO	Aalto University
DLR	Deutsches Zentrum für Luft- und Raumfahrt
CAIRE	Cooperativa Architetti e Ingeneri - Urbanistica
Rovereto	Comune di Rovereto
TSB	Berlin Partner for Business and Technology
Tampere	City of Tampere
Logica	CGI Suomi Oy

LIST OF FIGURES

1. INTRODUCTION

As written in the DoW, WP4 is to deliver a control panel for public administration and political decision makers to understand the current traffic and associated emission situation and to influence it to provide means for effective emission control. Therefore a control loop consisting of data analysis and visualization, simulations and applying policies and strategies modelling needs to be implemented.

At this early stage of the project this deliverable is a preliminary view on the specifications and requirements to fulfil the objectives of this work package. For each of the four tasks of this WP the specifications and requirements are described, and selected tools and mechanisms are evaluated. It might come out that because of risks, which are described later on, some decisions taken by now will not fit best over the whole time of the project. Those will be adopted by time and reconsidered again.

2. TASKS

WP4 consists of four parallel tasks. This section describes the requirements and specifications for each task. Therefore the pilot sites described in deliverable *D6.1 Specifications of city pilots for the first STREETLIFE operation and evaluation* are considered and were taken into account. This is to ensure that the results of WP4 are matching to the pilots at the end of the project. The following descriptions of requirements and specifications were evaluated to fulfil the use cases and generated requirements of the pilot sites.

2.1. T4.1 Global data view & analytics

2.1.1. Requirements

There are two major types of requirements to create and populate a global view of the traffic and carbon emission: Section 2.1.1.1 deals with the data needed to fulfil the task and Section 2.1.1.2 with respect to the tool chosen to visualize and analyse the data.

2.1.1.1. Data requirements

The following data sources are required to visualise the global traffic and carbon emission situation.

- The (motorized) *street traffic density* (vehicles/km^2) at some level of granularity, the finer the better. Optional, and if available: the (average) speed, travel time, and number of stops of the travels.
- The *public transport activity* (actual capacity/km^2) and usage (actual number of passengers/km^2) at some level of granularity, the finer the better.
- The current volume of *bike traffic* (e.g. as a density in terms of bikes/km^2).
- The *pedestrian activity* (e.g. as a density in terms of pedestrians/km^2).
- Real-time information about *events* (e.g. disruptions, accidents, cultural events), including the location of the events.

- If available: CO_2 measurement data.

The actual data will be provided by WP3 / T3.1. Requirements with respect to the *data format* will ultimately depend on the visualisation tool to be chosen for this task (see below).

2.1.1.2. Tool requirements

The envisioned visualisation tool must be able to properly process and visualize all the data listed in the previous section. This comprises the following issues:

- Plotting data on a street map in (near) real-time:
    - Colouring of (polygonal) areas on the map (given by the density data),
    - And / or colouring of street segments (if the density data is given in terms of street segments),
    - Plotting symbols at specific locations (visualisation of events),
    - Plotting text messages at specific locations (visualisation of events).
- The possibility to define *user-defined functions* (UDFs) to implement key performance indicators (KPIs).
- Plotting KPIs (scalar values) in gauges, as bars or as colour indicators.
- Visualisation of historic data:
    - Plotting *cumulated* information (as specified above) for a (user-defined) period of time on a street map,
    - Plotting historic KPIs (UDFs) in a time-value graph for a (user-defined) period of time.
- Ability to import data in their given format, also streaming data.

The visualisation tool must be chosen in accordance with the above-mentioned requirements. There are tools available, which will be presented later on. A further examination of these tools is required to decide which tool fits best and if and how it's possible / necessary to modify the tool to fulfil the mentioned requirements.

2.1.2. Specifications

The following items specify the envisioned features and functions of the STREETLIFE global traffic monitoring system:

- (Near) real-time visualisation of all density data (cf. data requirements) on a street map by
    - Colouring areas on the map according to the given density ("heat map"),

   - And / or colouring of street segments according to the given density (if the density data is given in terms of street segments).

- (Near) real-time visualisation of all event data (cf. data requirements) on a street map by
   - Plotting symbols at specific locations (visualisation of events / alerts),
   - Plotting text messages at specific locations (visualisation of events / alerts).
- Plotting KPIs (scalar values) in gauges, as bars or as colour indicators.
- Visualisation of historic data:
   - Plotting *cumulated* density information (as specified above) for a (user-defined) period of time on a street map,
   - Plotting historic KPIs in a time-value graph for a (user-defined) period of time.

The actual availability of the individual functions will depend on the available data provided by WP3 / T3.1.

2.1.3. Visualisation tools

The following visualisation tools have been examined with respect to their suitability for the STREETLIFE global traffic monitoring system. They all have capabilities of displaying data in various ways and can be used within the project. We need to take into account that might not only one tool resolve all issues, we probably need to combine different tools with each other.

ITEF [10]: ITEF is a tool for the visualisation of microscopic traffic simulations or the evaluation of real car traffic experiments. It is developed by Fraunhofer FOKUS. In particular, it can be used to visualise and replay recorded travel routes. It has no support for the visualisations specified in 2.1.1. and 2.1.2. Moreover, it does not support real-time monitoring. Thus, in its current form it is not suited for STREETLIFE task 4.1.

CIP [3]: CIP from Siemens AG is a platform for analysing city-related data. The platform offers dashboards to visualize various information including geo-based topologies, historical data, forecasting data, and so forth. Other analytics tools including TIBCO Spotfire from TIBCO Software Inc. can be connected to CIP to perform additional analysis of city data. CIP supports the visualization of street maps using Google Maps. CIP, in its current status, does not support real-time data streams. Processing of real-time data is planned to be implemented in CIP. The other parts of the requirements in 2.1.1., plotting gauges, bars, and graphs, can be implemented via CIP dashboards.

Google Maps API [6]: With the Google Maps API one can embed Google Maps in web pages or mobile apps. In its standard version (free of charge), a service using the Google Maps API must be freely and publicly accessible to end-users. The commercial version of Google Maps, called *Maps API for Business* does not have this restriction and supports 100 000 requests per 24 hour period, as opposed to 2500 requests permitted for the free version (as of Dec. 2013). Apart from these restrictions, the Google Maps API is in principle suitable for the first part of the requirements in 2.1.1.: plotting data on a street map. However, the Maps API is not a self-contained visualisation tool and requires some relatively low-level JavaScript programming. The other parts of the requirements, plotting gauges, bars, and graphs, need to be implemented by other means.

OpenStreetMap [11]: OSM, to some extent the free counterpart of Google Maps, is a web community project to create and use a free editable map of the world. The data is available under the *Open Database License*, a "Share Alike" license agreement intended to allow users to freely share, modify, and use the database. It can be used through JavaScript libraries (*Leaflet, OpenLayers*) or libraries for other programming languages, e.g. Ruby, R, or Python. Like the Google Maps API, OSM is not a self-contained visualisation tool and it requires some relatively low-level programming. Also OSM is in principle suitable for the first part of the requirements in 2.1.1. i.e. plotting data on a street map. The other parts of the requirements, plotting gauges, bars, and graphs, need to be implemented by other means.

In summary, there is currently no single visualisation tool known to fulfil all the requirements given in 2.1.1. An integration of OSM into CIP could be a possible solution, or, as another alternative, a pure low-level implementation in JavaScript using either the Google Maps API or OSM, plus other libraries for plotting gauges, bars, and graphs.

2.2. T4.2 Simulation based data forecast

2.2.1. Requirements

Required input data: The following data are needed to set-up and perform the simulation scenarios defined by the STREETLIFE simulation use cases.

- Road network data: road network including specific data; e.g. maximum speed, number of lanes, etc.
- Road infrastructure data: infrastructure data relevant for the traffic flow; e.g. traffic light phases, electronic traffic signs, roadside units, etc.
- Individual driver data: data to model each relevant vehicle individually; e.g. origin-destination matrices or complete travel routes of vehicles.
- Further input data might be necessary depending on specific use cases not defined so far.

Simulation tool requirements: The following requirements have to be fulfilled by the simulation tool.

- The simulation tool has to provide a microscopic traffic simulation: Since the individual behaviour of STREETLIFE users is to be analysed, an individual model of the STREETLIFE user is needed.
- The simulation tool has to provide a modelling of mobile STREETLIFE applications and central STREETLIFE components.

- The simulation tool has to provide a modelling of the communication among mobile STREETLIFE applications and central STREETLIFE components.
- The simulation tool has to provide a modelling of the produced emissions.
- Further requirements might have to be fulfilled depending on specific use cases not defined so far.

Simulation scenario requirements: The following requirements have to be fulfilled for the specification of simulation scenarios:

- Detailed specification of simulation use cases
- Specification of metrics and KPIs to be analysed by simulations
- Specification of mobile STREETLIFE applications (e.g. smartphone apps) used to inform / interact with the STREETLIFE users
- Specification of central STREETLIFE components (e.g. traffic management centre components) used to generate / provide the information needed by the STREETLIFE users

Evaluation requirements: The following requirements have to be fulfilled for the analysis / visualization of simulation results:

- Specification of parameters / metrics to be analysed
- Specification of needed map-based visualizations (e.g. visualization of congested areas)
- Specification of needed diagrams to analyse the dependencies of different parameters / metrics

2.2.2. Specifications

For the specifications of simulation scenarios and evaluation details, the above requirements (simulation scenario requirements and evaluation requirements) have to be fulfilled. Since the preparation of the corresponding definitions is still in progress.

2.3. T4.3 Mobility policies and strategies modelling

2.3.1. Requirements

The main objective of a sustainable mobility plan, and the policies linked to it, is the rationalization of the movements of things and people within the territory of the study. Mobility is influenced by two sets of elements:

- Places where the urban functions are located, represented by houses, workplaces, the services, which generate the demand for mobility. These areas are identified in relation to the prevailing functions within the urban fabric. In a more generic way these places can be interpreted as places of origin or destinations of daily trips that take place within the territory.

- The infrastructure whose task is to serve trips of the various components of urban traffic. According to the national guidelines for urban mobility, the components of traffic that must be taken into account are, in order of importance: the pedestrian mobility, mobility with collective public transport, mobility using private car and the parking of private motor vehicles.

The key to the interpretation of the territorial model is constituted by the correlation between these two sets.

The first input is used to study the characteristics of mobility, or more properly the different conformations of the mobility demand, and therefore represents the organization of the urban disposition. The identification of the strategic elements inside mobility policies has to be applied to the relational tissue in order to highlight the connecting elements between the settlement system and mobility system.

The interpretation of settlement pattern is followed by the definition of the functional classification of roads in centres and in communication networks. The ranking of the roads expresses, in fact, the role of each road related to its main use, generated by the function to which that road gives answers in the functional settlement, mainly with respect to the type of connection that the road guarantees to commercial activities and residential facilities, both of production and tertiary, and functional poles properly said.

Below, there is a list of the possible mobility policies that could be implemented in the three pilot sites. At this point of the project the list is preliminary. STREETLIFE at its best should be an optimal instrument for mobility managers to choose between ranges of options regarding mobility policies.

2.3.1.1. Policies for regulation of motor vehicle movement

Possible policies in the field of info-mobility are an increase of the panels with information, or a system that not only gives information to the end-users but also manages to collect information from them.

All possible useful information for the end-users are potential requirements in this kind of policies. Once there is a good level of knowledge on the actual situation, that level can be augmented with other useful information, and this information can be very heterogeneous, from the operation of a Traffic Limited Zone to the location of parking lots, or availability of services, and so on.

2.3.1.2. Parking policies

Parking policies involve different aspects of city management: from pricing, to the location of parking supply, to land planning, with connections also with mobility information and the impact of real-time information of the usage of parking lots.

Requirements needed for this group of policies are the location and capacity of parking lots, the fares applied, and their usage rate of different times and days of the week.

2.3.1.3. Policies for innovation of public transport

Another way of impacting the mobility habits is improvement on public transport. Improvements for public means of transportation can be achieved in many ways: modernization of vehicle fleet, improvement of the network, more user-friendly pricing, communication and marketing, and a better organization for tourists transport.

In this type of policies is required a deep knowledge of the public transport system: routes, timetables, fares, type of vehicles used.

2.3.1.4. Policies to promote walking and cycling

Policies to promote slow mobility involve land planning (like reorganization of public space and continuity of the cycle-pedestrian network), but also mobility information and expansion of bike-sharing services.

Information needed for this type of policies includes the maps of cycling roads, and number and capacity of bike sharing services and bike-parking structures. As a baseline bicycle flows and bicycle distance are useful information. A map with the location of cultural and commercial polarities could also be very useful to check out how the slow mobility network relates to those polarities.

2.3.1.5. Policies for urban logistic and management of fleet operating

Policies on the goods movement are always the hardest to implement: in order to organize the distribution of goods to certain places or obtain a sustainable presence of commercial vehicles the information needed involve the distribution in time and space of goods movement, and it’s not an easy task to get all the info needed.

2.3.1.6. Policies for mobility management

Policies involving mobility management take into account car sharing, the marketing of tourism based on sustainable mobility, and coordination of decision-making processes that lead to the implementation of mobility management policies for the employees of the centre’s activities.

2.3.1.7. Policies for the mobility information

Possible policies in the field of info-mobility are an increase of the panels with information, or a system that not only gives information to the end-users but also manages to collect information from them.

All possible useful information for the end-users are potential requirements in this kind of policies. Once there is a good level of knowledge on the actual situation, that level can be augmented with other useful information, and this information can be very heterogeneous, from the operation of a Traffic Limited Zone to the location of parking lots, or availability of services, and so on.

2.3.1.8. Policies for the events management

Policies for events management should help users to reach and leave the event without waste of time and / or energy. Requirements for these policies regard the offer side, like time and place of the event, available parking capacity and means of public transportation or car / bike sharing available, but also on the demand side, like the estimated attendance of the event and the estimated source of people coming to attend it, and also the estimated duration of the event itself.

2.3.2. Specifications

All the types of mobility policies expressed in the former section will be injected into a rule system language, to enable STREETLIFE to simulate the behaviour of vehicles on the roads and extracting the best possible scenarios for the mobility system of the pilot sites.

An expert system is a computer program that represents and reasons with knowledge of some specialist subject with a view to solving problems or giving advice.

To solve mobility problems at the expert level, expert systems will need efficient access to a substantial domain knowledge base, and a reasoning mechanism to apply the knowledge to the problems they are given. Usually they will also need to be able to explain, to the users who rely on them, how they have reached their decisions.

Expert systems can be developed with expert system shells. An expert system shell is a software-programming environment, which enables the construction of expert or knowledge based systems. Expert systems software can be developed for any problem that involves a selection from among a definable group of choices where the decision is based on logical steps. Any area where a person or group has special expertise needed by others is a possible area for an expert system.

2.3.2.1. State of the art

The base for the technology that has also been written in the DOW may consist of an expert system rule language, but after a first review of the possible solution, for our project seems a better fit the usage of a Business Rule Approach [1] for the following reasons:

- BRA fits better for abstract policies and strategies
- BRA is an external source → rules are used by, not embedded in, management systems (expert systems are typically closed, are designed to solve an entire problem on their own)
- The rules should be readable by managers and technologists
- Rules are generated by rule discovery technologies (like decision trees, data mining, big data analytics…)
- Rule management technologies and rule execution technologies are mostly combined in business rule engines
- The goal of rule execution is inference, it is divided into inductive inference and deductive inference
- Some rule engines can establish their conclusions (how and why was the rule founded?)

Figure 1 Process chain of expert systems

2.3.2.2. Survey on available techniques

The list of the following techniques is divided into rule discovery tools and rule engines. The following list consists of tools, which are capable of fulfilling the requirements. Again it is a combination of different tools. Each mentioned tool is still supported and widely used.

2.3.2.2.1. Rule Discovery

HyperCube [7]: Hypercube is capable of processing huge quantities of data. The strength of this algorithm lies in its ability to identify homogeneous groups that outperform the average and explain why they behave in a different way.

Hypercube already has outputs in the form of business rules, and is one of the best software available at extracting models and rules from big sets of data

RapidMiner [13]: RapidMiner is another very useful tool for data mining and as a consequence of its history RapidMiner enables database import from many formats, including Excel, Access, Oracle, SQL, SPSS, and also text files and Dbase.

RapidMiner has tools that enable the user to analyse every size data source, and it is a user-friendly software: it has open extension APIs to develop own algorithms, it contains a batch mode for utilization without GUI, and the outputs can be exported in various formats, like XML, XML zipped, or binary.

WEKA [15]: Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from a java code. WEKA is an open source software, which can import data from ARFF, CSV, Binary, URL, Database (using JDBC).

Orange [12]: Orange is a data mining software package, whose functionality can be accessed through Python, and is commonly used for bioinformatics.

2.3.2.2.2. Rule Engines

Drools [4]: Drools is a platform for business rules management and complex event processing. It is compatible with Rule Machine Learning, contains a java library, and it works on Eclipse-based IDE or is available in maven repository.

Jess [5]: Jess is a rule engine and scripting environment written entirely in Java language. Jess is small, light, and one of the fastest rule engines available. Its powerful scripting language gives access to all of Java's APIs. Jess uses an enhanced version of the Rete algorithm to process rules. Jess has many unique features including backwards chaining and working memory queries, and of course Jess can directly manipulate and reason about Java objects. Jess is also a powerful Java scripting environment, from which you can implement Java interfaces without compiling any Java code.

Microsoft Business Rules Engine [2]: The Business Rules Framework is a Microsoft .NET compliant class library. It provides an efficient inference engine that can link highly readable, declarative, semantically rich rules to any business objects (.NET components), XML documents, or database tables. Application developers can build business rules by constructing rules from small building blocks of business logic (small rule sets) that operate on information (facts) contained in .NET objects, database tables, and XML documents.

IBM Operational Decision Manager [8]: Previously known as Jess Rules, it is composed of two components: the IBM Decision Center that provides an integrated repository and management components, and the IBM Decision Server whom provides the runtime components to automate decision logic

Intalio | BPMS [9]: Intalio is the first Open Source product for Business Process Management. It enables integration with different Form and UI technologies and gives also the possibility to use an expensive web service API

Visual Rules [14]: Visual Rules is a business rule engine designed for the highest level of performance. It is particularly well suited for performance-critical applications that require rapid results (e.g. online transactions, mass data processing). Sequential algorithms accomplish the processing of rules. Visual Rules provides the option of employing business rules as decision services across the company's systems, including Service Oriented Architectures (SOA). Additionally, services can be orchestrated within Visual Rules: external services can be called from within the rules. Rules are delivered as executable Java code with Visual Rules - e.g. for offline execution within mobile utilities.

2.4. T4.4 Mobility management and emission control panel

This task defines a mobility management and emission control panel and aggregates the results of the other tasks of WP4 into one platform.

2.4.1. Requirements

The control panel has to aggregate all other task results to a single platform. The software outcomes of the other tasks have to be implemented in a common technology or they must have well defined interfaces to communicate to each other. It needs to react on changes and events in real-time and has to process a huge among of data.

2.4.2. Specifications

The control panel is the single entry point for the user to interact with the visualization, the simulation and the creation and editing of the model policies. Each component is defined and implemented in its task, but all is wrapped in the control panel. The control panel acts as a platform navigating the user through the workflow of:

- Visualize data
    - Map based
    - Charts
- Monitoring of KPIs
- Interacting with events like warnings for certain areas
- Generate correlations between different kinds of data
- Manage models and policies for simulation
- Perform simulations based on applied models
    - Micro and macro simulation
- Compare simulation results regarding KPIs before and after simulation
- Provide well-defined web based interfaces
- Realization as a web based platform.

2.4.3. Tools

Possible tools acting as a basis for the control panel are:

- **FOKUS Open Data Platform:** The Open Data Platform is a catalogue for metadata describing open data. In its current status it's not able to store the data itself, but this is planned for next releases as well as the visualization of the stored data. Due to its portal technology[1] it can easily be used as a per user customizable dashboard.
- **FOKUS Data Cloud:** The Data Cloud is able of storing data through various kinds of formats. Communication to the Data Cloud is realised via web-based interfaces. All data are searchable via SQL queries, even across different datasets. As the Data Cloud runs on cloud infrastructure it provides a sandbox for JavaEE based web services for third party developers like PaaS. So far real-time capabilities are not tested.
- **Siemens CIP** (see section 2.1.3)

3. WP4 ARCHITECTURE APPROACH

This section gives a short description of the available component and data given by each pilot site, which has a connection to WP4 activities. After that challenges will be described and a first architecture approach for a solution will be explained.

[1] JSR-000286 Portlet Specification 2.0 (https://jcp.org/aboutJava/communityprocess/final/jsr286/index.html)

3.1. Pilot site state of the art

3.1.1. Berlin

3.1.1.1. Available components

Within the Berlin city region different stakeholder (Car Sharing Companies, Bike-Sharing Companies, Bike-Taxis, public transport provider, a traffic information centre and the City of Berlin) possess numerous data sets relevant for the project objective. These data sets are necessary for the project and efforts should be put into negotiations with the data providers to make them available to project. Then, they additionally have to be analysed towards the project requirements.

3.1.1.2. Available data

Beside the data sets mentioned above, there are a small number of sets already available for the public. Here, the City of Berlin has make steps forward to open data for the public and specifically for app developers. Berlin Open Data is the project, which currently offers more than 300 data sets within 24 categories.

Two sets very relevant for the project is the public transport timetable for the 1st halve of 2014 and the charging stations for electric powered mobility, which are presented below.

- Public Transport Timetable 1/2014

Currently the public transport provides an interface for the Berlin-Brandenburg region to make bus, trains and light rail timetable available to the customer. The data are in the GTFS-format[2]

- eMobility charging stations in Berlin

The city of Berlin provides a list of geo-coordinated charging stations for electric powered mobility.

3.1.2. Tampere

3.1.2.1. Available components

The Tampere pilot site offers a lot of components that are useful for client applications, e.g. routing planner for bicycle tours and multi modal trips. An interesting component for a control panel is the bus traffic real-time monitor.

3.1.2.2. Available data

In Tampere lots of data are available, most of them as open data or open services. The formats of the data are in a good shape for machine readably operations. Regarding WP4 the Tampere pilot site offers data for:

[2] https://developers.google.com/transit/gtfs/).

- Road and street network
- Public transport
- Route planning
- Street lights
- Traffic flow
- Road conditions
- Traffic Incidents
- Weather
- CO_2 emission

The data for public transport are available in real-time.

3.1.3. Rovereto

3.1.3.1. Available components

The simulator that currently is used for the traffic model in Rovereto is proprietary software by Citilabs Inc. called CUBE. The software is a simulation and analysis tool for city traffic and mobility, which covers the functionalities of Macro Simulator and adaptation engine.

The software needs a license fee, and CAIRE owns one. This software must be considered as a background for the STREETLIFE project, and will be used as a black box, although means of integration with the rest of the STREETLIFE software deployed in Rovereto will be studied.

3.1.3.2. Available data

We have collected various types of data for the Rovereto Pilot, and we classified the information into various groups: maps, busses and trains, parking, traffic and viability.

The first group is <u>maps</u>: this group includes the city road network, other information on the road network like numbers and the names of the roads and streets. In this kind of data we have also pick-up points related to bike sharing and car sharing.

The second group of available data regards <u>Buses and Trains</u>, so there are all the sources connected with Public Transport, like routes and timetables of buses, as well as timetables and delays of trains.

The third family of available data that will be part of the model of simulation is about <u>parking</u>: on the city of Rovereto we have disposal of street-level parking data, and parking garages availability.

The fourth group of available data that will be part of the model of simulation is about traffic and viability: information on road accidents and traffic jams, data about car flows and bike flows.

3.2. Challenges

Each of the pilots has different scenarios and use cases, some of which other pilots do not have. Some of these components are already finished and available for some pilots, others are not. But almost all of the pre-existing components do not lie in the thematic priority of WP4. They refer to a more specific level in the pilot sites whereas a mobility management and emission control panel in the focus of this WP addresses a higher management related level. Remaining available components as well as available data regarding to WP4 are listed in section 3.1 above.

Hence, WP4 it is quite unrestricted to define architecture for a mobility management and emission control panel. Therefore, the goal of this definition process should be a common platform for this management and control panel placed over STREETLIFE pilot sites, instead of developing single or hybrid modules for each single pilot. Furthermore, the WP4 architecture has to be aligned in this process with the general STREETLIFE architectural blue print in WP2.

In the mobility management and emission control panel different WP4 relevant topics have to be addressed. For emission control CO_2 calculations are necessary. Management policies have to be projected and park and ride as well as car and bike sharing have to be supported also regarding to modal split. Additionally, KPIs from the different pilot sites have to be derived, aggregated and monitored in the control panel. These KPIs might be partly pilot specific but it is not excluded that they cannot be calculated for other pilots.

The interface to the pilots can be a generic interface. This interface has to connect traffic management systems and trip planning in the different pilots with the mobility management and emission control panel. In order to influence trips e.g. by policies the trip planning has to be addressable before and after the planning process in the trip planner.

If it is possible existing pilot APIs are used. Otherwise an API as an interface for the WP4 control panel has to be developed. This is one open question that has to be clarified during the architecture process in the next steps. It applies also to the different formats, interfaces, licences and ownerships for existing data and components. For usage in WP4 the possible field of application has to be cleared. E.g. for the existing components in the Tampere pilot the legacy system needs to be taken into account in the architecture. For the Rovereto pilot most software components have free accessibility for STREETLIFE partners. The same applies to the available software components in the Berlin pilot.

Another open point concerning the data storage has to be clarified. Does the mobility management and emission control panel in WP4 need an additional data model and data storage e.g. for KPI calculation and the management of calculation results? This has to be decided during the on going architecture process. Furthermore, a potentially critical point might be the communication to external service providers that should be addressed but are not in the STREETLIFE project.

These open questions and maybe further questions that occur in the next development steps have to be answered in the architecture process.

3.3. Architecture conclusion

In section 3.2 it was already observed that it is possible for WP4 to freely define an architecture for a mobility management and emission control panel due to few existing and restricting software components in the pilots. In the following, a first architecture approach is presented.

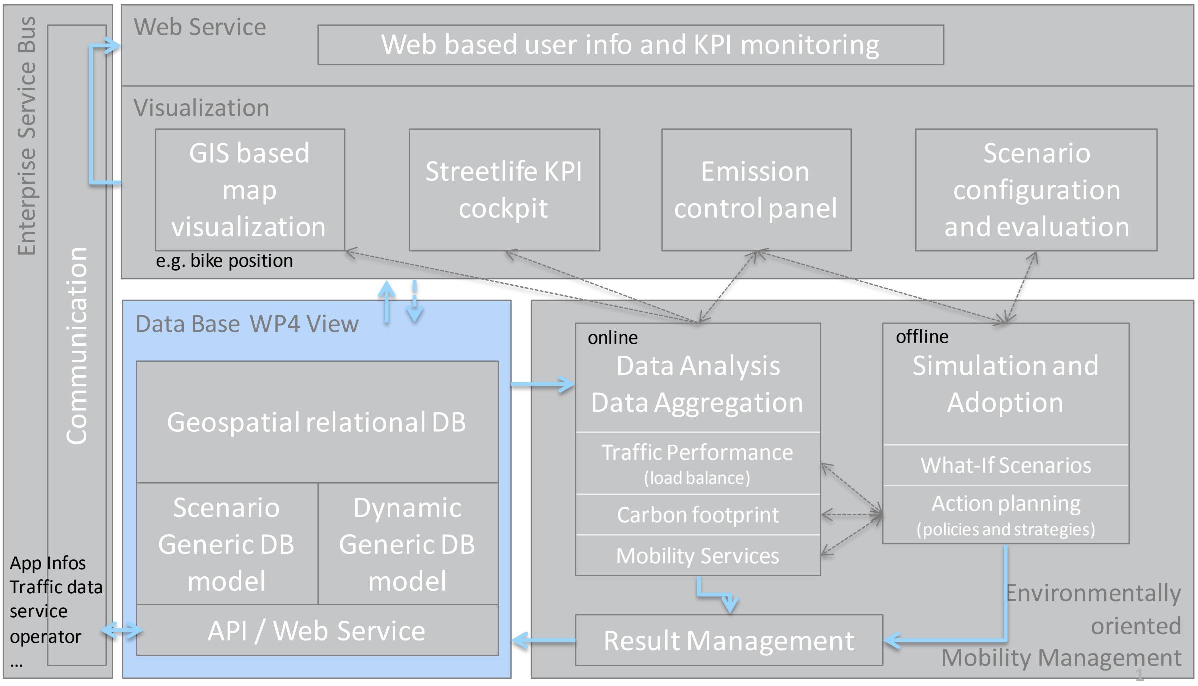

Figure 2 WP4 functional architecture approach

Basically there are four modules with different functionalities, see Figure 2:

- Environmentally oriented mobility management
- Data base for WP4 aspects
- Visualization (local and via web service)
- Enterprise service bus for communication

Every module fulfils specific high-level functionality.

Environmentally oriented mobility management:

- Data analysis and data aggregation for online applications to show traffic performance, carbon footprint and mobility service coordination. Input for these evaluations is dynamic data from (see also section 2.1)
    - Traffic management about traffic situation and resource utilization
    - Traffic and transportation affecting events
    - Sensor data about e.g. air quality
    - Routing and actually used transportation modes from STREETLIFE App
    - Service operation about occupied and available transportation modes (cars, bikes, electric vehicles,..) as well as parking opportunities including localization

Output is result information for visualization and communication

- Simulation and adoption for offline decision support with what-if scenarios and action planning (see also section 0 and 0).
  These what-if scenarios might be e.g. micro traffic simulation starting from actual traffic situation and short term forecasting impact of special events to define counter measures for operation mobility management on the one hand, and macro urban simulation for modelling long term policies and strategies e.g. for impact evaluation of environmentally oriented mobility services to support easy switch from individual motorised transport to public transportation modes on the other hand.

Result management handles the different results and prepares them for addressed users (e.g. App routing, service operator, mobility management centre, web service,...)

Database for WP4 aspects:

- Geospatial relational database for object information management, geo referenced relation evaluation and geo referenced visualization (coloured mapping) using static and dynamic standardized data.

- Dynamic generic data model used for collecting all dynamic input data from different sources (e.g. STREETLIFE App, mobility management, public transportation, service operation, sensors,..) and different location (pilot sites). To use the collected data as a platform input for application specific evaluation there must be a defined application interface.

- Scenario generic data model is used for what-if scenarios, which means the storage of defined input data, like topology information and scenario specific parameterization and configuration as well as management of result data and time lines for comparison of different alternatives.

- API or web service is necessary for interactive data access.

These data base aspects are WP4 specific but should be integrated into the overall STREETLIFE architecture and database, especially for data integration from other work packages like WP3 and WP5. As mentioned in section 3.2, data storage for WP4 aspects has to be clarified.

Visualization: (see also section 2.1)

- GIS based map visualization is needed for specific applications for mobility management, like localized visualization of traffic densities via colour code, or availability of shared bikes and cars, or positioning of events and additional information.
- STREETLIFE KPI Cockpit visualizes the aggregated mobility performance in diagrams and scalars for benchmarking, historic progress and trend development.
- Emission control panel shows especially the carbon footprint and impact of changes.
- Scenario configuration and evaluation is used as graphical and control user interface for offline scenario management.
- Web based user information and KPI monitoring via web service is used for STREETLIFE web site, to inform STREETLIFE community about daily achievement but also for STREETLIFE performance analysis. The public information is aggregated form local visualization modules.

Enterprise service bus:

- Communication within enterprise service bus allows generic data access for all data sources and data sinks over all pilot sites

4. CONCLUSION

This deliverable shows the requirements and specifications of each main component of WP4. This is a very early estimate to fulfil the objectives of this WP.

Section 2.3 shows a lot of generic policies that need to be taken into account when integrating the components of an expert system. Also in section 2.3 we described tools for the process chain of an expert system. These tools will be evaluated in more detail in the next steps, to find a capable and integrated solution for the expert system in WP4.

Section 3.1 shows that there is a need for a control panel in Berlin, Rovereto and Tampere. So for the next steps WP4 will focus to come up with solutions for Berlin and Rovereto. Considering the available components shows a significant different for the simulation part of WP4. As there are data (TAPAS) and VSIMRTI as a simulation environment for micro simulation available for Berlin. Rovereto uses the CUBE macro simulator and the experience of CAIRE for its scenarios.

The evaluation of the components and data available for each pilot shows that there are different aspects of the WP4 control panel covered in some pilots and missing in others. To solve this issue a modularized solution was presented in section 3.3. The presented architecture is a first approach and need to be discussed with other work packages. At least the pilot sites, WP2, 3 and 5 needs to be aligning with this approach. As it turns out that the Siemens CIP platform is a potential candidate for a basis of the WP4 technology stack a very early next step is to go into technical details of this platform to evaluate how new components can fit to the CIP technology.

APPENDIX A: LITERATURE AND SOFTWARE COMPONENTS

[1] Ross, R.G. Principles of the Business Rule Approach. Addison-Wesley, 2003.

[2] Business Rule Engine. Microsoft, http://msdn.microsoft.com/en-us/library/aa561216.aspx

[3] City Intelligence Platform. SIEMENS, http://www.siemens.com/innovation/apps/pof_microsite/_pof-fall-2013/_html_en/city-intelligence-platform.html

[4] Drools - The Business Logic Integration Platform. Red Hat, https://www.jboss.org/drools/

[5] Friedman-Hill, Ernest. Jess, http://www.jessrules.com/

[6] Google Maps API. Google, https://developers.google.com/maps/

[7] HyperCube. BearingPoint, http://hypercube.bearingpoint.com/

[8] IBM Operational Decision Manager. IBM, http://www-03.ibm.com/software/products/en/odm/

[9] Intalio Bpms. intalio, http://www.intalio.com/products/bpms/overview/

[10] Intelligent Test Evaluation Framework. Fraunhofer FOKUS

[11] OpenStreetMap (version 1), http://www.openstreetmap.org/

[12] Orange: Data Mining Toolbox in Python (version 2.7), http://orange.biolab.si/

[13] Rapidminer. rapidminer inc., http://rapidminer.com/

[14] Visual Rules. BOSCH, http://www.bosch-si.com/products/business-rules-management-brm/visual-rules.html

[15] WEKA (version 3). Unviversity of Waikato, http://www.cs.waikato.ac.nz/ml/weka/