FP7-SMARTCITIES-2013

STREETLIFE

Steering towards Green and Perceptive Mobility of the Future

WP4 - Mobility Management and Emission Control Panel

D4.2.2 – Mechanisms and Tools for Mobility Management and Emission Control (intermediate)

Due date: 30.09.2015 **Delivery Date**: 04.10.2015

Author(s): Benjamin Dittwald, Carsten Schmoll, Jens Kohlmorgen, Björn Schünemann, Silke Cuno (Fraunhofer), Birgit Obst (Siemens), Davide Frigeri, Francesco Avesani (Caire)

Partner(s): Fraunhofer, Siemens, DLR, Caire, VMZ

Editor: Benjamin Dittwald (Fraunhofer)

Lead Beneficiary of Deliverable: Fraunhofer

Dissemination level: Public **Nature of the Deliverable:** Report

Internal Reviewers: Tom Schilling (VMZ), Frank Behrendt (BPWT)

EXECUTIVE SUMMARY

This deliverable is based on the results that has been presented in D4.2.1 in Y1. It concentrates on the changes that has been made since the last deliverable and references to it if necessary. Most changes apply to the Mobility Management and Emission Control Panel (MMECP) UI and the simulations. It shows how most of the announced scenarios are integrated into the MMECP.

Disclaimer: This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 608991.

Mechanisms and Tools for Mobility Management and Emission Control (intermediate)

Table of Contents

ABBREVIATIONS

API	Application Programming Interface
BER	STREETLIFE Berlin-Pilot
CO	Confidential, only for members of the Consortium (including the Commission Services)
CIP	City Intelligent Platform
D	Deliverable
DoW	Description of Work
FCD	Floating Car Data
FP7	Seventh Framework Programme
FLOSS	Free/Libre Open Source Software
GWT	Google Web Toolkit
GUI	Graphical User Interface
HLA	High Level Architecture
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IPR	Intellectual Property Rights
ITEF	Intelligent Test Evaluation Framework
KPI	Key Performance Indicator
MGT	Management
MS	Milestone
MMECP	Mobility Management and Emission Control Panel
MC	Modal Choice
MVC	Model-View-Controller
NUTS	Nomenclature of Territorial Units for Statistics
OS	Open Source
OSS	Open Source Software

OSM	OpenStreetMap
O	Other
OD	Origin / Destination
P	Prototype
PU	Public
PM	Person Month
R	Report
ROV	STREETLIFE Rovereto-Pilot
RTD	Research and Development
TAPAS	Travel Activity Patterns Simulation
TRE	STREETLIFE Tampere-Pilot
WP	Work Package
Y1	Year 1

PARTNER

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 Ingegneri - Urbanistica
Rovereto	Comune di Rovereto
TSB	Berlin Partner for Business and Technology
Tampere	City of Tampere
Logica	CGI Suomi Oy
VMZ	VMZ Berlin Betreibergesellschaft mbH

LIST OF FIGURES

LIST OF TABLES

1. Introduction

This is the intermediate version of the "Mechanisms and tools for mobility management and emission control" deliverable. The here presented work is based on the experiences from the initial version of this document. As the initial version provided the vision of the MMECP, we now have started with the development and can present our results.

The key results are directly related to the tasks of Work Package 4.

T4.1 Global data view & analytics - The MMECP generic interface was extended with meta information in order to provide better visualisation capabilities, see section 4.

T4.2 Simulation based data forecast - The current status of the simulations for Berlin and Rovereto will be shown in section 5.

T4.3 Mobility policies and strategies modelling - The scenarios described in section 3 are more concrete now, as they are shown in the MMECP directly. Not all of them are included in the MMECP yet, but will be in Y3.

T4.3 Mobility management and emission control panel - The MMECP has brand new UI and technology stack. The new UI and what has changed till Y1 will be shown in section 2

2. New MMECP UI background technology

During the last year, the MMECP UI was heavily improved. A lot of work has been done to make the UI cleaner, more intuitive and more feature complete. This section is about the changes that has been made to the implementation of the MMECP UI.

During the last Year 2 of STREETLIFE when developing the MMECP User Interface (UI) one job was to exchange the technology the UI was built upon. Initially in STREETLIFE Y1 we used Google Web Toolkit (GWT, see [1]) for the interface, because it looked promising for the MMECP UI. GWT is a Google web framework. We went for it, because its supports map based applications well. It offers the developer to implement client side applications in Java and converts them into JavaScript. Practically, it turned out that GWT is not easy to maintain when used in real-time environments. That is why in Y2 we transferred to AngularJS, see [2]. Another reason for the change was, that during our developments, the Google GWT started to support Java 7 instead of Java 8, which we need. Java 8 is still not supported by GWT in the moment. This and the fact that some libraries used by GWT are outdated, limits the options a developer has to implement applications.

The change of background technology was possible since the STREETLIFE MMECP is designed according to the Model-View-Controller (MVC) principle. This design principle gives us the chance to re-implement the UI, without affecting the backend of the MMECP.

Besides that, preparing that change, our work consisted in the evaluation of more promising frameworks:

- JavaServer Faces [3]
- Wicket [4]
- SenchaJS [5]

- AngularJS

We decided for usage of AngularJS, because it comprises a cleaner structure than GWT, is easier to maintain, fits well with our real-time requirements when used with web sockets and has a great support of OpenLayers, the map framework we use for the UI. We have good experiences with AngularJS in other projects, though.

Figure 1 shows a screenshot of the old UI. Figure 2 shows a screenshot of the new STREETLIFE MMECP UI and uses red rectangles to highlight and describe its main components.

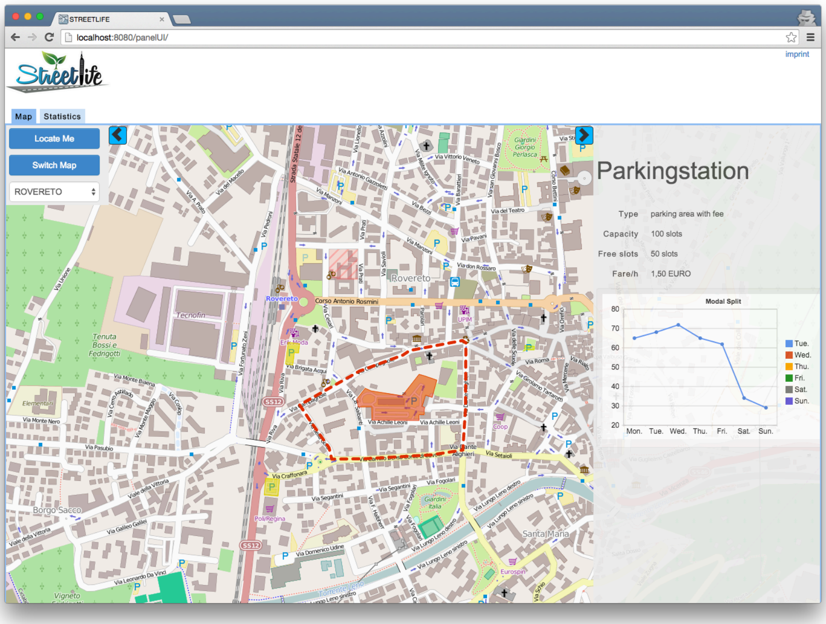

Figure 1: Old MMECP UI – STREETLIFE Y1

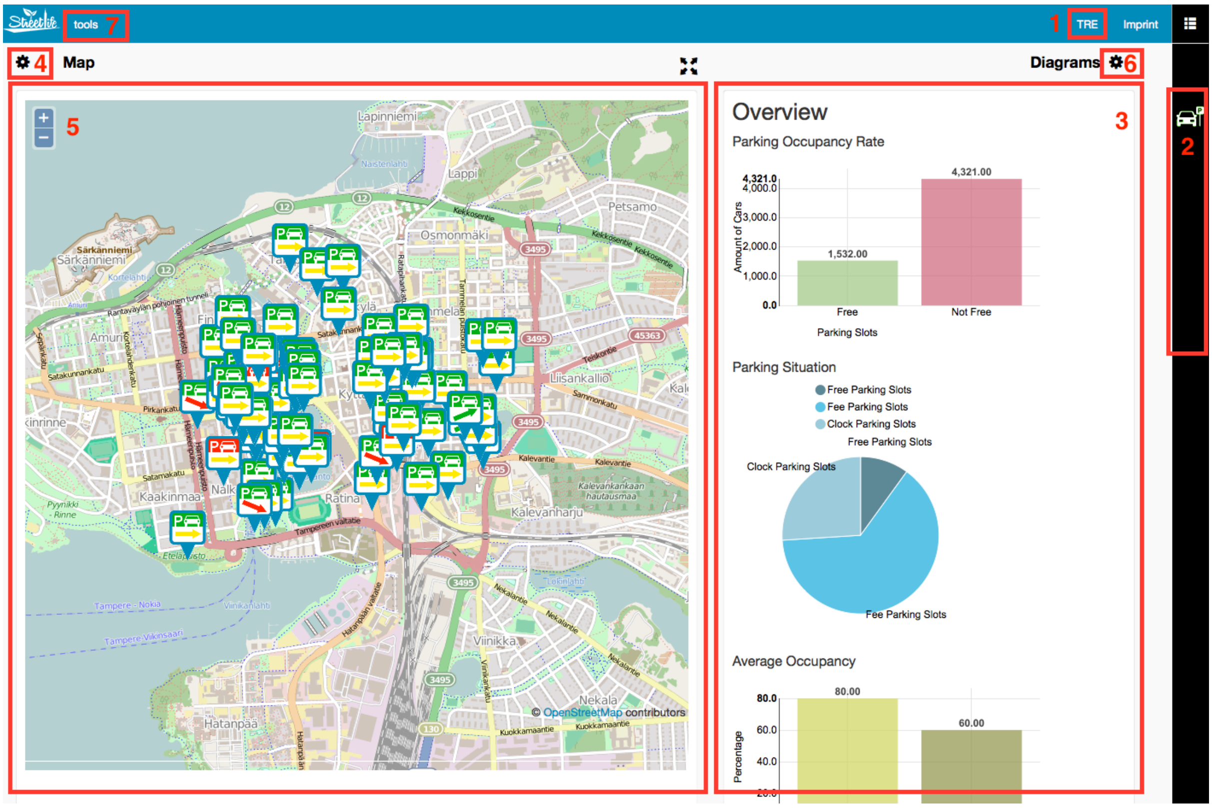

Figure 2: Main components of the MMECP UI

No. 1 indicates a drop down menu to select the desired city on the monitor. Actually, usage of this function through a mobility manager is not contained in any scenario, but such use case is imaginable. In case a mobility manager is responsible to monitor more than one city, he can use the menu to quickly jump to the desired city. During the STREETLIFE developments this menu is used to test the MMECP-functions across the STREETLIFE city pilots. Though, it might be possible

No. 2 right hand bar contains a list of scenarios that are available for each city. They are indicated as icons. The scenarios were described in the former Y1 Deliverable 4.2.1 [6] and will be illustrated again for better comprehension in in this deliverable. The selected pilot city Tampere shown in the current screenshot in Figure 3 above has until now realised only one scenario for the MMECP. This is the reason why you see only one icon at No.2 - the icon for the “Park & Ride scenario”.

No. 3 is an area in the interface that is used to show diagrams that give a global information overview of the information contained in the selected scenario. In the screenshot, the diagrams show some statistics about the parking lot situation of Tampere, since the Park & Ride scenario is selected.

A central part of the MMECP UI is its map view (No. 5). The map view is able to show different layers of data that can be selected in the layer selection (No. 4). In the case of the Tampere pilot, it uses the live-data for occupancy of the parking lots. A more detailed description of this scenario is in section 3.33.3.

Some scenarios allow the user to apply additional filters to the objects, shown on the map. For instance, timely restrictions. These filters can be selected with a form that can be shown by clicking on 6.

Another approach of filtering is the use of geographic filters. By clicking on the *tools* label (7), some painting tools can be used to draw rectangles, polygons or ellipses on the map. Afterwards the user can apply actions to the geographical selections.

What is still missing in the current version: Time and geographical filters are not implemented yet. These are functions planned to be ready at the beginning of 2016. To run these filters the “STREETLIFE Common API”, described in Section 4 of this Deliverable will be used. Drawing is supported by OpenLayers and AngularJS.

3. Refined scenarios of the Y2 version of the MMECP

In the ongoing “climateprotection dialogue”, in which the city of Berlin is figuring out measurements for CO_2–levarage, first results published in a feasibility study 2014 titled "Klimaneutrales Berlin 2050"[1] showed that the future action strategies and measures in Berlin towards CO_2-reduction relate to five action fields: These action fields are: energy supply, building and urban development, economy, households and consumption and traffic.

The goal of the ongoing public dialogue kicked off by the study is to institutionalize the topic and to release in near future clear plans and laws for Berlin. During a recent meeting[2] of the climate protection dialogue STREETLIFE partner VMZ pointed out in a presentation in front of the Senate of Berlin the three adjusting screws in the action field “traffic”: These are: fleet division, fuel consumption reduction and modal split. With the change of the modal split – they mean reduction of cars, rise of ecological friendly transport (bikes, pedestrian, public transport) and especially more bicycle usage is seen as important challenge for Berlin. For this alternative mobility offers have to be extended: Cycling, public transport and walking has to get more attractive to bring about a behavior change of the traffic participants. Mobility management will play a central role in there!

Through this official statements the STREETLIFE project feels confirmed to be on the right track! The “Modal Split”-change is seen as a key factor and important CO_2-crank in the “action field” traffic. The statement within the public climate dialogue confirms STREETLIFE selection of scenarios, as they focus on Modal-Split towards more bicycle usage - in case of the MMECP –component - through a set of measurements of effective traffic-management comprising monitoring, forecasting, simulations functionalities focusing in the scenarios directly towards bicycle capacities, cyclists safety, Park & Ride facilities - all of this very relevant for a carbon-friendly modal split.

In case of the cycling in the climaprotection dialogue it was worked out so far for cycling following measures:

[1] See: The dialogue and the publications can be downloaded here: http://www.stadtentwicklung.berlin.de/umwelt/klimaschutz/bek_berlin/.

[2] At the 2. Stadtdialog Energie und Klimaschutz” at the 28.09.2015.

- Creation of secure parking possibilities at major destinations of leisure, shopping and commuter traffic.
- Improve storage facilities at bus stops
- Further expansion and arrangement of bicycle strips and bicycle paths throughout the main road network
- Improve the parking situation and options at route intersections
- Provide secure routing information and secure cycling infrastructure

The STREETLIFE scenarios described in the following are in-line with that.

Besides that in Year 2 technical improvements of the MMECP took place– which was mainly in the exchange of technology for the UI. The MMECP scenarios to be shown have been updated as well since last Year. Also one new scenario was added to last year's portfolio. The following section describes the scenarios listed by their main functionalities which are: Forecasting, CO_2-Monitoring, Simulation.

The Y2 evolution of the scenarios is described and elaborated how the scenarios suit into the current version of MMECP.

3.1. Forecasting

The forecasting mechanism is a central part of the MMECP, though it is not fully included yet into the UI, its backend improved significantly in Y2. This is shown in the following sub-sections. For the forecasting the MMECP distinguishes momentarily between the scenarios "Event Attendees and Modal Split Prediction" and the "Bike Availability Prediction", both scenarios will be used in Y3 during the Berlin Pilot test.

3.1.1. Event Attendees / Modal Split Prediction

Relates to: Use Case ID: BER-MGMT-4

The forecast mechanisms built into the MMECP-system is supposed to be used by an event manager in cooperation with the Transport Service Operator. It is constructed to estimate the required transport capacities per means of transport (bus, bike, walking, car …) for a large upcoming public events. Commonly, they can then decide on the most suitable mode of transportation. The projected needs are high enough to mandate a higher service schedule (e.g. more buses, bring extra bikes to places near the event …). Ideally, some means to receive feedback after the event allows the evaluation of the quality/correctness of the proposed measures. This mechanism was described in more detail in D4.2.1, see [6].

3.1.2. Bike Availability Prediction

This chapter describes the bike availability forecast implemented going needed for the management of big social events. The service application is used within the Berlin management use case BER-MGMT-3 for information transparency of bike availability at shared bike stations in general and especially for prediction of bike availability at social events at "Tempelhofer Freiheit", as first approach for guidance.

The former old city Airport "Tempelhof" from Berlins thirties was recently shut down and converted due to the decision of citizen participation into a park. The new "Tempelhof Park" was made available to the public as an event and recreational area in the Berlin districts of Neukölln and Tempelhof. It lies on the Tempelhof Field of Teltow plateau, covers 355 hectares of the site of the former Tempelhof Airport, making it actually Berlin's largest city park. The Tempelhofer Park is part of the project "Tempelhofer Freiheit", which is comprising further the long airport building and the apron of the former Tempelhofer airport, which was constructed in the time of National Socialism.

Data model for information transparency and prediction

The data used for analysis, forecast and visualization is stored in an oracle data base within CIP (City Intelligence Platform) and is continuously updated. As available data we use information about the social events, shared bike stations and weather situation.

Actually a list of social events taking place at the Tempelhofer Freiheit within June 2014 and November 2015 is used. It includes the modal split of transportation modes the visitors use for their arrival and departure to the events. Furthermore we use information about the bike availability at bike stations. Providers of such service are CallABike and NextBike in Berlin. They provide data or bike station status and update every 30 minutes. We can use it for historical data analysis and for prediction related to the social events. The weather information is intended to be used for correlation issues but the algorithm is not realised yet. The weather data is drawn from "openweathermap" and "wunderground" data sources.

The data model for data access is defined as followed:

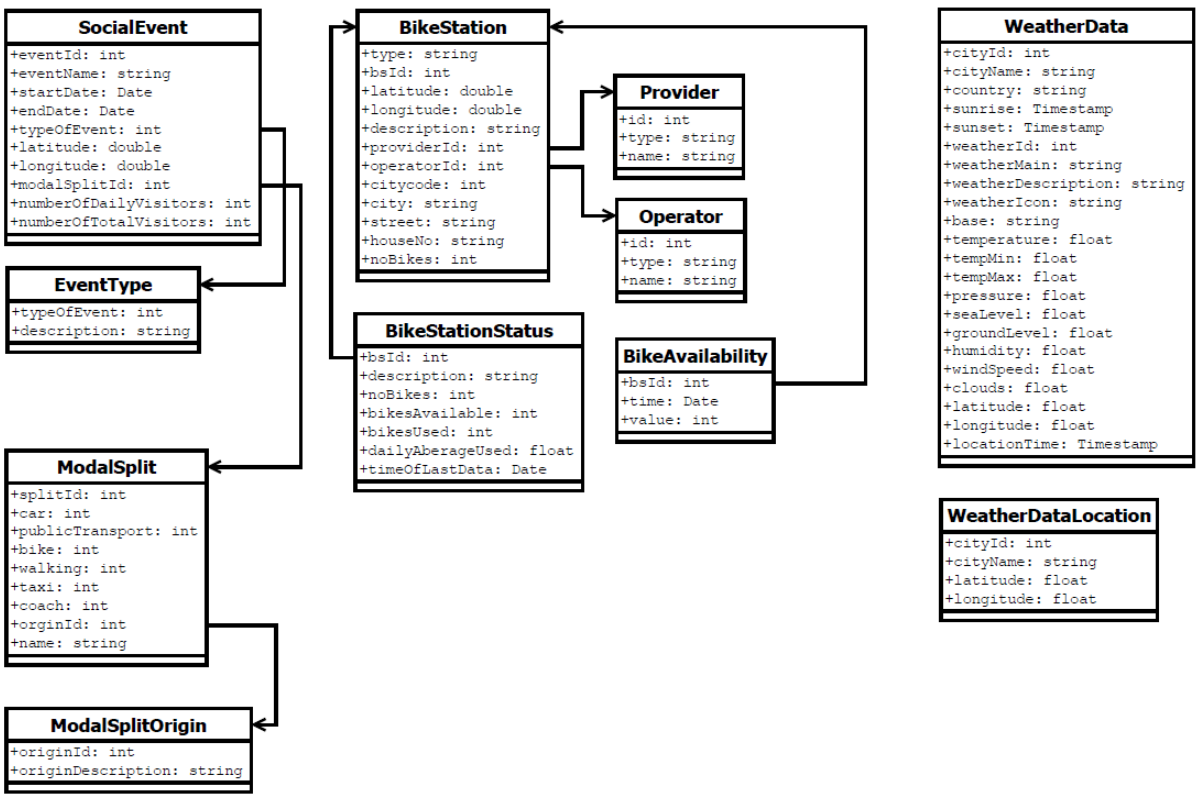

Figure 3: Data model for social events, bike availability and weather

Service Requests

The access to stored data and evaluation results is realized by REST services, which are deployed in the CIP. Actual implemented back end services are listed in Table 1:

Table 1: Bike availability forecast API description

service	description
Modal Splits	Lists all modal split patterns (data model *ModalSplit*) defined for social event types to give information about the percentage split of transportation modes used by visitors of social events.
Event Types	Lists Ids and description for all defined types of social events (data model *EventType*)
All Events	Lists the information of all social events (data model *SocialEvent*) stored in the data base
Single Event	Get the information for a single event (data model *SocialEvent*), selected by attribute *eventId*
Events by Type	Lists the information of all social events (data model *SocialEvent*), selected by attribute *typeOfEvent*
Events by Type and Location	Lists the information of the social events (data model *SocialEvent*), with attribute *typeOfEvent* and at given location (*latitude, longitude*)
All Providers	Lists the information of all providers (data model *Provider*)
All Operators	Lists the information of all operators (data model *Operator*)
All Bike Stations	Lists the static information of all bike stations (data model *BikeStation*)
Bike Stations by provider	Lists the static information of all bike stations (data model *BikeStation*) from provider, selected by attribute *providerId*
Bike Stations by operator	Lists the static information of all bike stations (data model *BikeStation*) from operator, selected by attribute *operatorId*
Current Bike Station Status	Get the current status information (data model *BikeStationStatus*) for bike station, selected by bike station id *bsId*
All Current Bike Station Status	Lists the current status information (data model *BikeStationStatus*) for all bike stations
Current Availability	Get the current bike availability/number of free bikes (data model *BikeAvailability*) at bike station with bike station id *bsId*
All Current Availabilities	Lists current bike availabilities/number of free bikes (data model *BikeAvailability*) for all bike stations.
Current Day Availability	Get bike availabilities/number of free bikes (data model *BikeAvailability*) of the current day at bike station with bike station id

	bsId, starting at 0:00 of the current day, every 30 minutes, until the actual time
All Current Day Availabilities	Lists bike availabilities/number of free bikes (data model *BikeAvailability*) of the current day at all bike stations, starting at 0:00 of the current day, every 30 minutes, until the actual time
Profile at Specific Date	Get daily profile (48 objects) of bike availabilities/number of free bikes (data model *BikeAvailability*), every 30 minutes, from 0:00 to 23:30, from bike station with bike station id *bsId*, at specific date (*Date dd:mm:yyyy*)
Profile Between Dates	Lists daily profiles (à 48 objects) of bike availabilities/number of free bikes (data model *BikeAvailability*) every 30 minutes, from 0:00 to 23:30, from bike station with bike station id *bsId,* between start date (*Date dd:mm:yyyy*) and end date (*Date dd:mm:yyyy*)
Profile by Weekday	Get daily profile (bike availability over a day with 48 float values, every 30 minutes from 0:00 to 23:10) of average number of available bikes over all bike stations of the city for the given weekday
Profile by Weekday for Bike Station	Get daily profile (bike availability over a day with 48 float values, every 30 minutes from 0:00 to 23:10) of average number of available bikes for the given weekday at bike station with bike station id *bsId*

Prediction of the bike availability

For steering reasons we predict the bike availability at the location of the scheduled events. The prediction is based on historical bike availability profiles of previous social events that took place the same location, similar type of event, same day of the week at similar weather conditions or at least similar season.

The process is as follows: The computational module in the MMECP picks an upcoming social event to carry out the bike availability prediction. If this specific event is an event that lasts for several days, the prediction has to be split, to consider every day of the event separately. A bike availability prediction has to be set up for every separate day. The data information the for social event must contain the start time and the end time of every event day. Additionally, the weekday and the season of the day of event needs to be defined for further evaluation.

In the prediction the year is categorized in seasons, spring (March, April, May), summer (June, July, August), autumn (September, October, November) and winter (December, January, February). For evaluation of the bike availability we define an event area with square of 500m length and breadth around the event location, see Figure 4. We suggest that visitors of the upcoming event will use the bike stations within this event area. To analyse the bike availability needed for the planned event we calculate with the sum of available bikes within this event area.

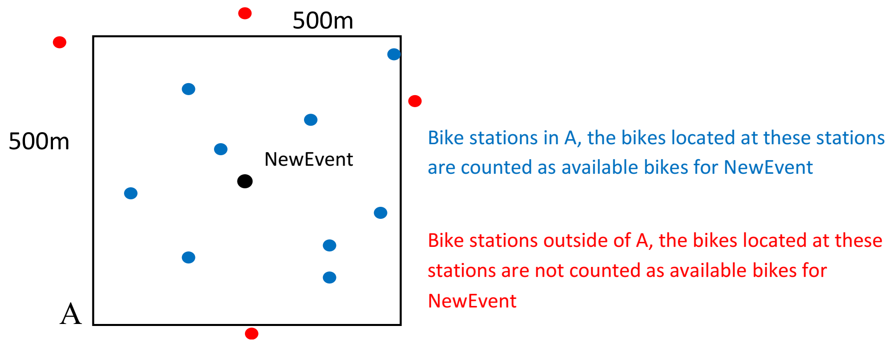

Figure 4: Area around an event

For prediction of bike availability for an upcoming event the pattern of average bike availability in this area with similar conditions but without an event will be evaluated; using week day categories, working day (Monday to Friday) and weekend (Saturday, Sunday) and season classes.

Next the bike availability in this area under similar conditions as previous events is used for evaluation of differences to the average pattern at start and end time; using additional categories for daily number of visitors (XS: very small event, visitors <100, S: small event: 100<=visitors<500, M: middle event: 500<=visitors<2500, L: big event: 2500<=visitors<20000, XL: very big event: 20000<=visitors) and types of social events.

The average factor of these resulting deltas (delta1 and delta2) is needed to predict the corresponding bike availability for the upcoming event.

A long term forecasting of the evaluated increase or decrease of bike availability caused by visitor arrival and departure at start and end of social events is necessary to influence the modal split positively. Timely forecast is needed by the traffic/event management to have precise figures on expected bike usage so there is enough time left to organised the necessary facilities and infrastructure regarding expected bike demand towards the bike operators and to pronounce traffic recommendations to the public.

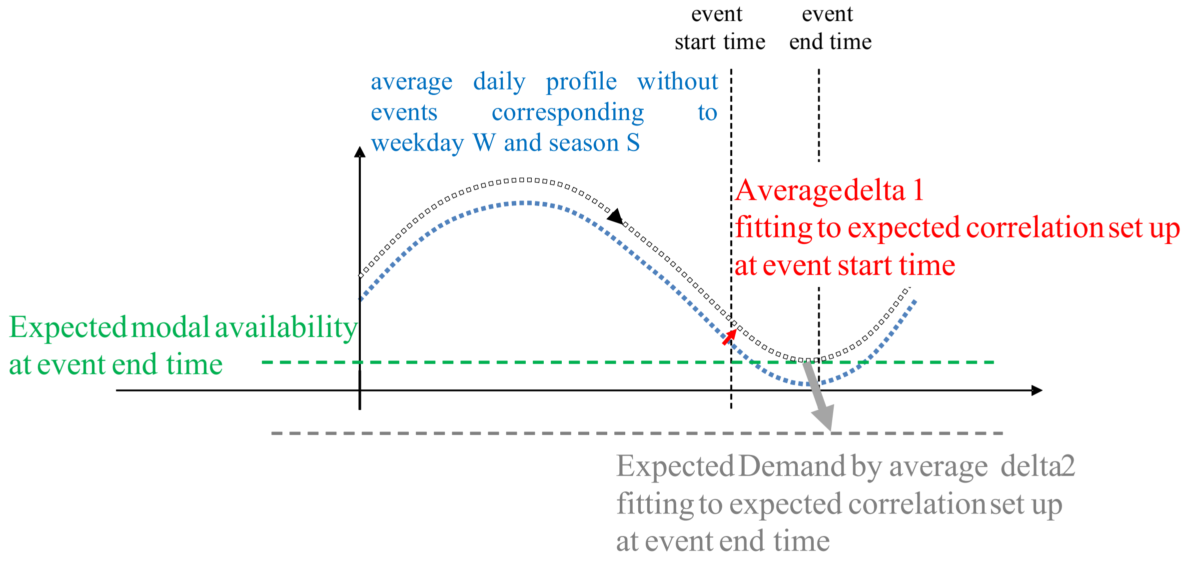

Figure 5: Long term bike availability prediction for social events

In the same manner short term forecast of expected bike demand is manageable, in this case it is based on actual measured bike availabilities of the event day.

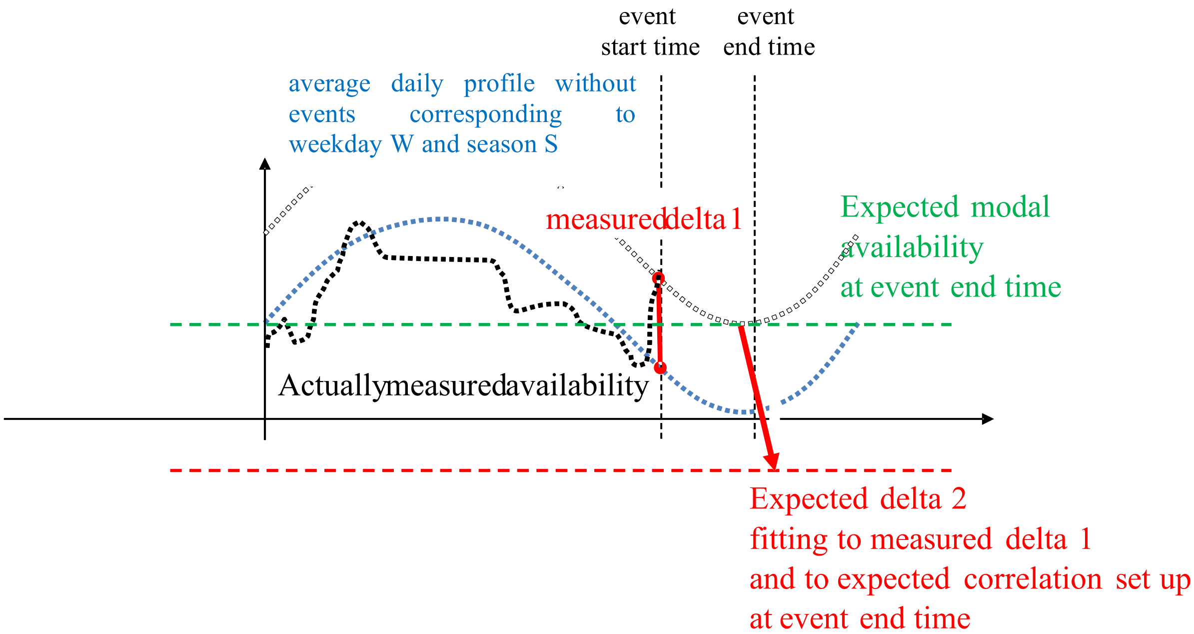

Figure 6: Short term bike availability prediction for social events

3.2. CO_2-Monitor

The CO_2-Monitor within the MMECP is a functionality to show of CO_2-situation in the area of Berlin.

CO_2-data: The available official "Open Data" concerning CO_2-Emissions for Berlin released by its competent organs can momentarily be described as poor. The official source situation of public available data in Berlin for CO_2 is as follows: Since 1998, CO_2-balances are for all countries in Germany published. They are created according to uniform rules and based on the energy balances. [3] Available are CO_2-emissions from the consumption of primary energy (source balance) *) are one figure per year by energy source for Berlin area in total: latest release is for the year 2012 (as of 07.27.2015). Also available are CO_2-emissions from the consumption of primary energy (source balance) *) per year by issuing sectors in Berlin: latest release is for the year 2012 (as of 07.27.2015). so in these cases you get exactly one figure and it is always at the city level and valid for a complete calendar year.

Interesting for the Emission Control Panel would be to have the official CO_2-emissions figures per day and on small scale level (streetwise). Such kind of data would be desirable as it would allow to identify the originators of the emissions and polluters and would ensure monitoring and transparency in the neighborhood about the real emission situation so direct measures could be taken. In technical terms this is no problem anymore, there seem to be are political barriers for the systematic collection and release of such data.

[3] See: http://www.lak-energiebilanzen.de/dseiten/co2BilanzenAktuelleErgebnisse.cfm.

As STREETLIFE doesn't have access to such data STREETLIFE had to find other ways to supply the Control Panel with data:

We identified three different approaches to be used to estimate the CO_2-Emmission more detailed. This section describes three approaches:

- Estimation via the STREETLIFE Mobility App
- Estimation vie the SrV2008 survey, see [7]
- Estimation via traffic data that comes from the Berlin VMZ

A conclusion about the three approaches closes this section.

3.2.1. CO_2-Estimation by using the STREETLIFE Mobility App

With the STREETLIFE Mobility App, data information about the trips from app users could be analysed and used for performance evaluation.

Out of the user model from D3.2.1 [8] following data model (Figure 7) could be used for KPI-evaluation:

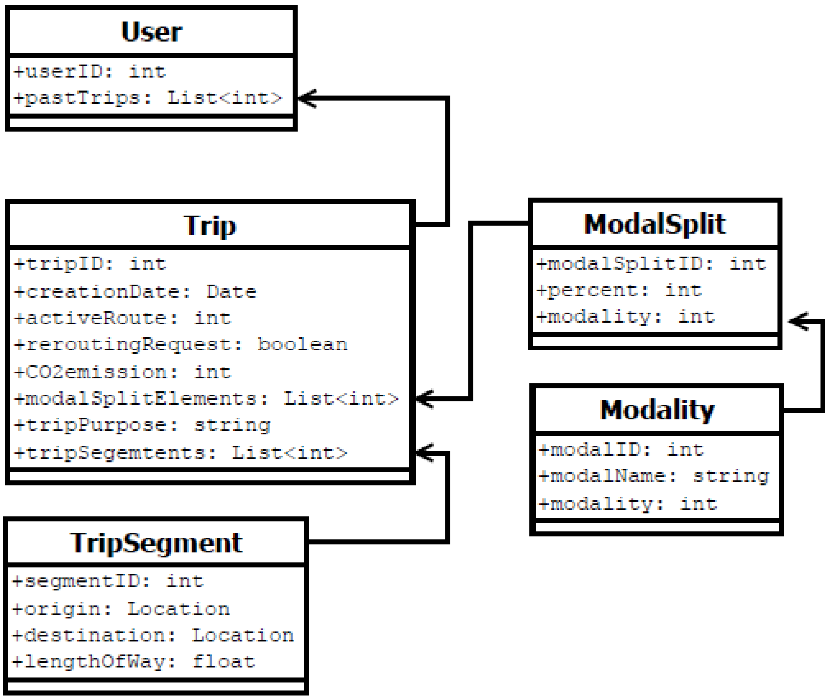

Figure 7: User model

These information performance parameters:

- Green Leaves
- CO_2-Emission
- ModalSplit

can be used and evaluated for the trips analyses by STREETLIFE Mobility App users.

An overview of hierarchy levels for evaluation of key performance indicator (KPI) range from data warehouse with static and dynamic data storage, analytics layer with calculation paths and objectives layer for result visualization, as shown here:

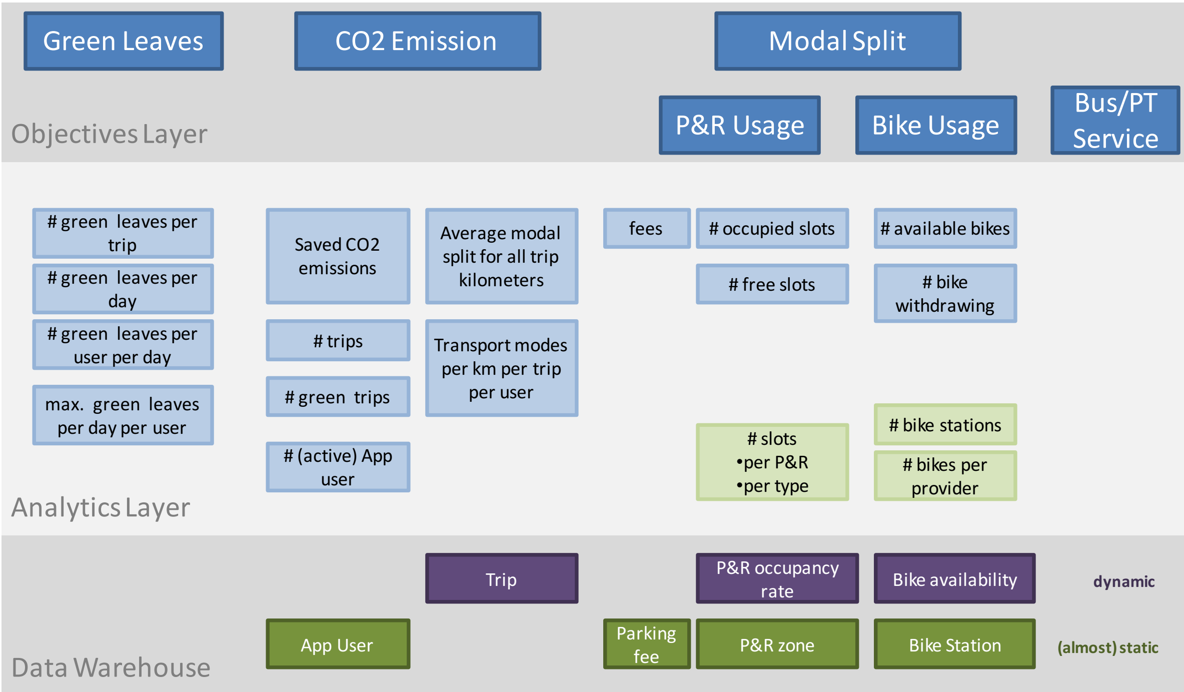

Figure 8: STREETLIFE KPIs and Indicators out of data storage

Using this data, the calculation of specific performance parameters is shown in this approach:

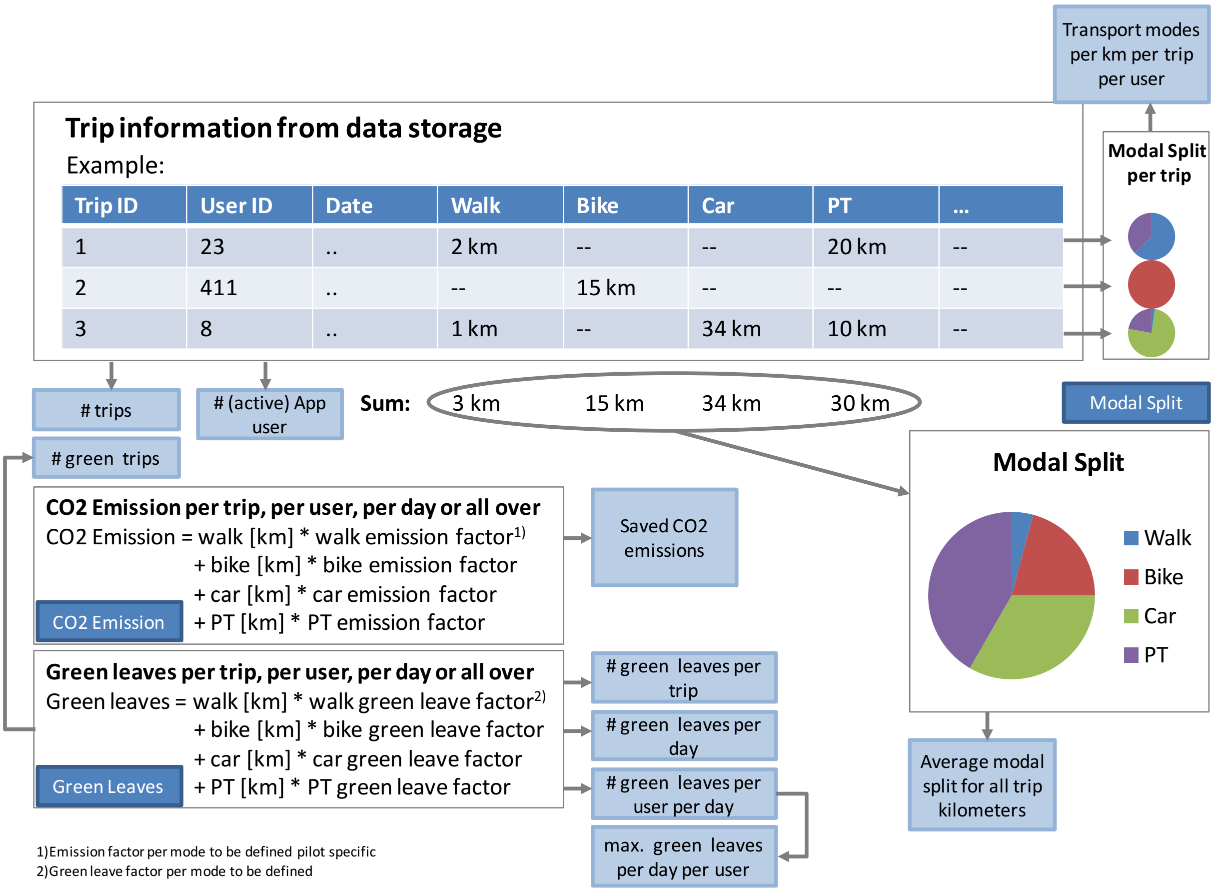

Figure 9: Trip calculations

3.2.2. CO_2-Estimation on the Basis of Srv2008 Survey

Another possibility to estimate the Berlin CO_2-Emissions is to consult the "SRV2008" [8] survey data. The Srv2008 survey was carried out on behalf of the Senate of Berlin. It was performed by contacting 40,000 citizens in Berlin. Its result is an extensive questionnaire filled by 35% of the investigated people. The survey was repeated in 2013, but the results are yet not public. STREETLIFE plans to investigate that too, in case it is released in time.

The 2008 survey results contain many trip information, including source, destination, and traffic mode used, which could be used to compute for example a static "heat map" over Berlin that forms the baseline of estimated mobility-induced CO_2-Emission in Berlin. The 2008 survey data is available to STREETLIFE but has not yet been fully evaluated due to the complexity of the survey's data structures.

The heat map could be drawn from information about source, destination and used transportation mode. It would give a qualitative output about the density of traffic across Berlin, based on the queried people from this survey. A simple heat map could look like Figure 10.

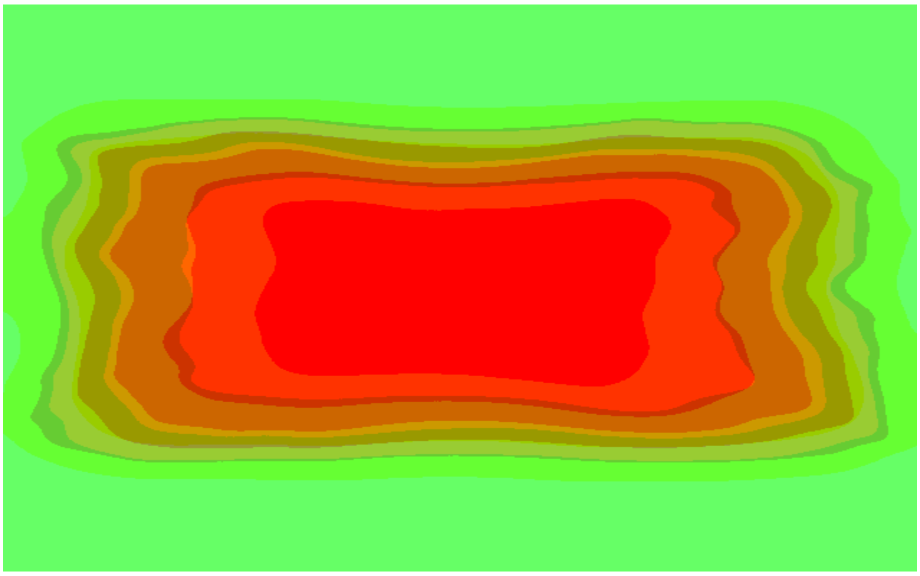

Figure 10: Simple CO_2 Example Heat Map

In a first step, one could assume that the CO_2-Emissions of a trip are generated across a direct line between travel source and target, and with a given, fixed width of the emission track.

Later, traffic routing could be used to "plot" the CO_2-Emissions of a travel along a path between source and destination that is possible and conceivable in Berlin in real life. This would make the computation more complex but the results more realistic. Care would need to be taken to route the path between source and destination only along realistic paths, e.g. not using the very shortest routes but those that would be realistic for cars and buses in Berlin.

The drawback of this approach would be that it only yields one new CO_2-Map for each SRV survey in Berlin, i.e. every 5 years. No real-time data is available from this approach. We still believe that it would give a valuable, realistic "base map" view, i.e. ground truth about CO_2-Emissions in Berlin and their distribution across city districts.

3.2.3. CO_2-Estimation on the Basis of Daily Monitored Traffic Data (by VMZ)

In Berlin, the VMZ, a central institution for mobility and traffic management, operates numerous traffic counters as well as sensors for environmental data. As a partner in STREETLIFE, VMZ can export assets of traffic-related information towards the backend of the MMECP. This data is based on vehicle counters mounted closely to traffic crossings and over big arterial roads of the city. For internal purposes, VMZ collects traffic statistics every 15 minutes. For data export to STREETLIFE, a somewhat lower interval will be targeted, in order not to interrupt the main operation of VMZ systems. STREETLIFE has granted access to an API that allows us to receive current data.

Based on this data we can roughly estimate the corresponding CO_2 emission. We measure the number of bypassing vehicles n_p per working day at a certain measurement point p. We also have an estimate of the average CO_2 emission rate of a vehicle, c, and use the average distance d between the measurement points to calculate the absolute emission of CO_2 between two measurement points for a single vehicle as follows:

$m = c * d$ *(in gram)*

We can then calculate the overall CO_2 emission for one working day related to a certain measurement point *p*:

$$m_p = m * n_p = c * d * n_p \text{ (in gram)}$$

In the STREETLIFE CO_2-scenario bases on this VMZ data, the operator is able to see the monitored values as differently coloured points on the city map, as an indication of very low (green colour) to very high (red colour) traffic density at each measurement point. Clicking on one of these points will display additional information about the sensor's readout about the rate of vehicles flowing through, below the sensor, in a fixed amount of time.

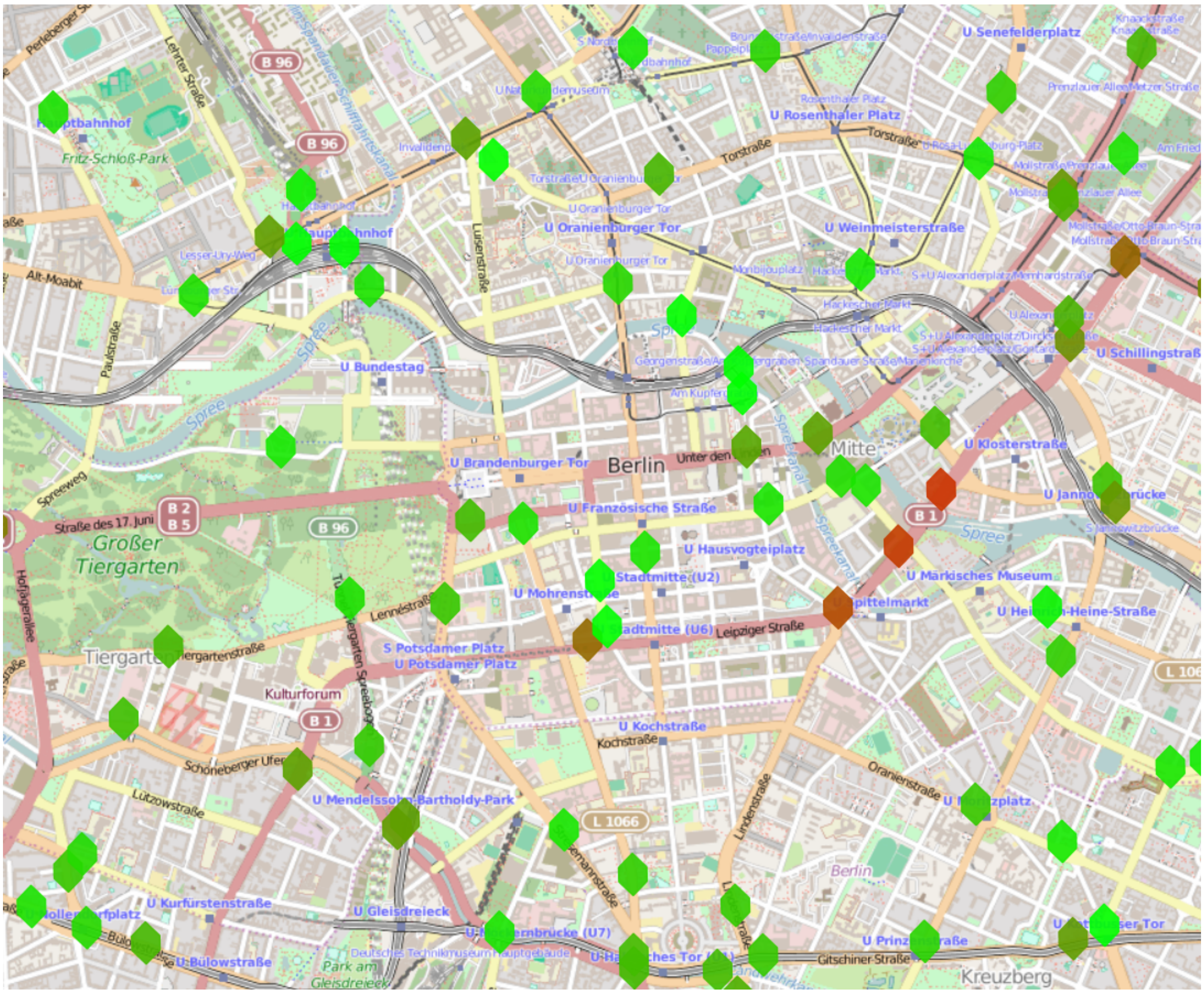

Figure 11: Berlin city map with Traffic Rate indication

In the future, the operator of the MMECP shall be able to select data sets from different times in order to compare current and past traffic loads.

A function for spatial aggregation of data points shall also ensure that the operator will not be overloaded viewing thousands of separate monitoring points if he/she zooms out of the map. Instead, the results of adjacent monitoring points will be merged for a zoomed out view.

Another function in a future release of the MMECP is to use this information in order to apply dedicated congestion charging actions to specific areas.

3.2.4. Conclusion

It can be stated that none of the above mentioned approaches can give an accurate estimation of the CO_2-Emission in Berlin. The data that is available momentarily to the project is too poor, in order to provide exact results on small scale. The mobility app for Berlin could provide enough information to calculate an accurate estimation if it has enough users that provide this data. We doubt that this will happen during the project lifetime. The SrV2008 is a very good basis for further calculations and to compare impacts, but the actual dataset available is unfortunately a couple of years old. It can be expected, that when the next dataset form 2013 is released to the public hopefully end of this year, this one again will lack in actuality, as it again will be 4 years old. The most realistic approach for STREETLIFE is to use the VMZ traffic flow data for CO_2-Estimation, as it is real and constantly updated data.

A discussion with a traffic manager working for the VMZ in Berlin on the idea to reflect CO2-values on small-scale onto a map of Berlin, best in real-time took place to review the usefulness and practicability of our scenarios. The VMZ explained that especially CO_2-values are difficult to be operationalised. That means it is difficult for the traffic manager to find and use short term measures for time wise affected areas to change a situation immediately. That is the reason why Berlin shortly is not planning to use CO_2-Calculations or Emission values on small scale in Berlin for the traffic management. CO_2-Measurements seem to them and to the city authority only useful for long term environmental measures and related planning. Such environmental measures are at least in the moment not the scope, of a conventional traffic manager.

During this conversation with the Traffic Management Berlin, it became obvious that the traffic management in a Smart City should be an “Integrated Management” which comprises the management of all single traffic modes and related data to provide measures in a climarelevant matter. In our STREELIFE vision of traffic management all traffic organizations exchange their information permanently and share a central steering platform using and exploiting also information from other relevant city sectors like for example environment and economy.

At least in Berlin the reality is different to the STREETLIFE vision. The traffic and the city mobility behavior are not permanent presented on screens to traffic managers like in its entirety and interconnectedness as envisioned and constructed in the STREETLIFE Emission Control Panel.

In Berlin the various traffic originations and transport operators like the S-Bahn, public city transport (BVG),the transport association VBB, etc. are working independently. They are acting as different legal and economic organizations and have various steering boards and control boards themselves. By now, it does not exit an overall an overall real time traffic steering in Berlin. In addition, the dedicated schooling and mandate of a traffic manager at VMZ is aligned solely to inform about disruption in public transport as well as for car traffic. There is no education in interpretation of traffic situations in the context of environmental measures – so in the moment it is not in the profile or task of the traffic manager to consider environmental measures.

The VMZ reported that an overall and collective traffic control for Berlin, in which all stakeholders collaborate together exist in Berlin solely as approach. Such an “integrated traffic control project” was indeed practised once before in Berlin in the case of the Football World Cup 2006. Here the different organisations worked together at a round table, the whole traffic situation was monitored through an helicopter from above, all had to process very

dense information series into coordinated information and acted for that special event in a very coordinated manner.
Such an integrated traffic control as permanent institution would be very desirable – especially in case of traffic bottlenecks and danger situations. In the moment the transport organisation – for example the underground – has to monitor and predict such situations themselves. If there is a situation of danger, they inform users themselves on the display, but even in such cases give little recommendations to customers to use alternative routes of other transport modes – The city is still not perceived as connected system. Here it the public sector is in a prime position to stimulate and promote the collaboration of a traffic organisation in the Smart City to put a full integrated traffic management in place.

3.3. Park and Ride

The Park & Ride scenario functionality in the MMECP has been designed with the goal to provide the city Mobility Manager a useful instrument to check and analyse strengths and weaknesses of the city's parking system. Once the mobility manager has a complete and full view of the performance of the parking lots in town, she will be able to implement policies and adjust the system to the city's mobility needs.

The default visualization of the MMECP in the Rovereto pilot section will focus on the parking system: on the city map the Mobility Manager is enabled to see at a first glance the occupancy level of different parking lots or city macro zones that include more parking facilities in one single area through their different colours.

Data visualised have been collected thanks to another Rovereto app developed into the STREETLIFE project: the CONTAPARCHEGGI app that has been used from traffic aides in order to detect in real-time occupancy rates for parking lots around Rovereto. The development at same time of the app and the control panel has been useful in order to create a geographic database format that is suitable for both devices, so that the transmission of data from the app to the MMECP has been comfortable and successful.

3.3.1. Description of the parking system and editing

The city map visualises the city parking system and the visualization can be selected for micro zones or macro zones. The geographic shape files for micro zones and macro zones are the same ones that are currently used in the CONTAPARCHEGGI parking aides' app.

The micro zones relate to the single parking facility: depending on the kind of information needed, the mobility manager can choose to see a) parking facilities where people are requested to pay, b) parking on the streets that require a fee, c) parking stations that allow users to stay only for certain periods, and the combinations of these type of parking spots simultaneously. On left side of the screen you find the different buttons that allow the Mobility Manger to choose the type of parking facility, the layer selection.

By clicking on the single object on the map, the mobility manager is able to see the data referring to the clicked object on the right side of the screen:

- Name and macro zone of the selected facility

- The fee zone (in the city of Rovereto there are different zones that have different parking rates per hour) and the fare per hour of the parking facility
- Type of parking slots in the structure and their number (e.g. slots with parking fee, slots with card block clock, slots reserved for people with disabilities)
- Average occupancy rate in the time band 10-12, that is the time of the day where usually there are the highest ones
- Graphic that shows the average occupancy rates for each day of the week

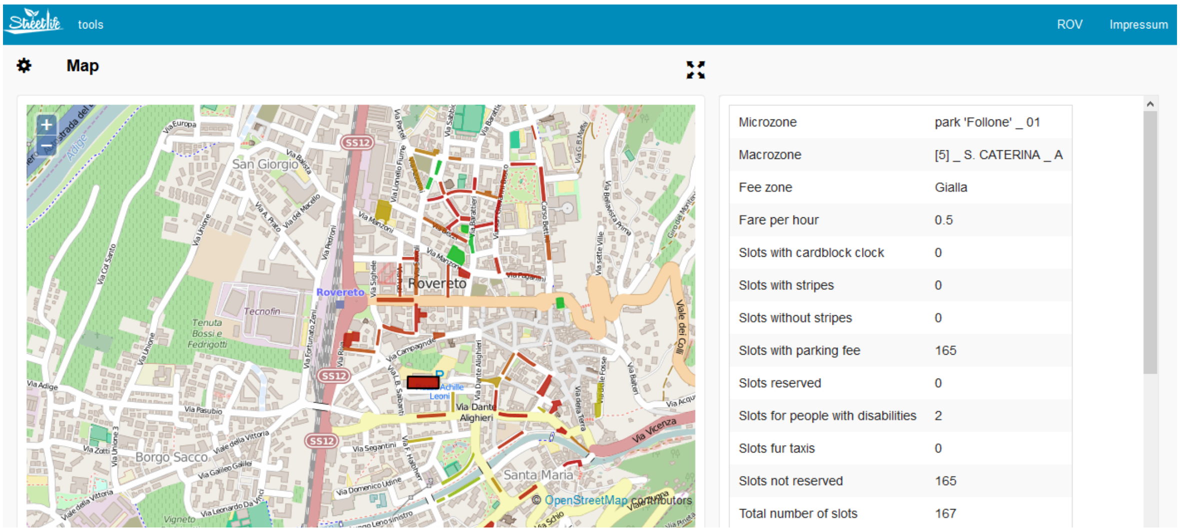

Figure 12: Description of a Parking Facility in MMECP

The visualisation shows city macro zones (or districts). It is a feature that aims at facilitating mobility manager's job: macro zones are parts of the city that are really useful for detailed analysis on the parking balance in different geographic parts of town, and represent the unit of measurement that is used to create reports and balances on the equilibrium between parking offer and parking demand. The macro zones feature is also an important functionality to make MMECP a useful instrument for the implementation of mobility policies, since the city Parking Plan use these geographical unit as the starting point for its analysis.

Macro zones data is based on the algebraic sum of the single parking facilities included in the delimited area, and the mobility manager can select on the left side the parking areas to see the state of macro zones. Like micro zones, also macro zones on the map are shown with different colours depending on their average occupancy rates. After clicking on the object, the mobility manager will see for the single macro zone the name, the number of measurements, and a graphic that shows the average occupancy rates for each day of the week.

Another functionality that is strictly related to the description of existing parking system is the editing feature (not still implemented in the existing MMECP). With the editing feature the mobility manager will be able to create new objects on the map or editing the features of the existing ones.

3.3.2. Analysis of the parking system

While the description of the system is more focused on the offer side of the parking system, the analysis functionality will give more details about the demand of parking slots in the city. All the occupancy rates detected through the ContaParcheggi App will from a historical database for each parking facility, parking on the street and every other type of parking facility in the city.

This historical database will be the source for the MMECP parking analysis: the user will select the geographic parameters (single parking areas or macro zones) and the time band for her analysis (which time of the day, which day of the week, which months and years), and will visualize on the control panel the dynamic of occupancy rates for the selected place and time. In the future development of the control panel it could produce a report on the balance of the parking system, that will show all the available data about offer and demand; these feature should be available only for macrzones, since for single facilities the current level of information showed on the screen is already enough.

A similar analysis feature that could be financially interesting for the municipality: The mobility manager could analyse and calculate the incomes from the parking system. As it already happens in the occupancy rates analysis feature, in this function of the MMECP the end user will be requested to provide an initial input regarding the desired geographic selection between a single parking facility or macro zones. After this initial input, the click on the selected area on the map will give to the mobility manager information regarding the fare zone of the city where the selected parking facility is located, the hour fee that exists in that zone, and the daily average income from parking fees in that area. This feature has more potential for its financial amount of information, but indirectly will give to the mobility manager also information about policies and the most used parking facilities.

The historic database on occupancy rates will enable the MMECP to create and update also the access cost indicator. (see D.4.2.1 p.62, [6]) The system will provide for each routing request the access cost indicator, that will express in minutes the estimation of extra time required in order to find an available parking spot, and this indicator is in correlation with the expected occupancy rate of the macro zone where the routing request will send the end user. The access cost indicator could become a useful instrument for the implementation of policies, since the mobility manager in particular circumstances could change it with the goal of patrolling the parking demand.

On a graphic standpoint, the actual interface is easy to use: on the left side of the screen there are buttons and features that help the mobility manager to choose place and time for its analysis, on the centre of the screen there’s the map with colours showing the different data, on the right will take place the statistical part, with graphics and table on the selected items.

3.3.3. Park & Ride scenario in Tampere

The Park & Ride scenario is basically the same as for Rovereto, but it features some important differences. Like in Rovereto it is used to measure the performance of the current parking lots. Compared to Rovereto, Tampere uses automatically observed parking lots, which provide real time data to the STREETLIFE system. Figure 13 shows a screenshot of the Park & Ride scenario in Tampere and the information that is available for the selected map object.

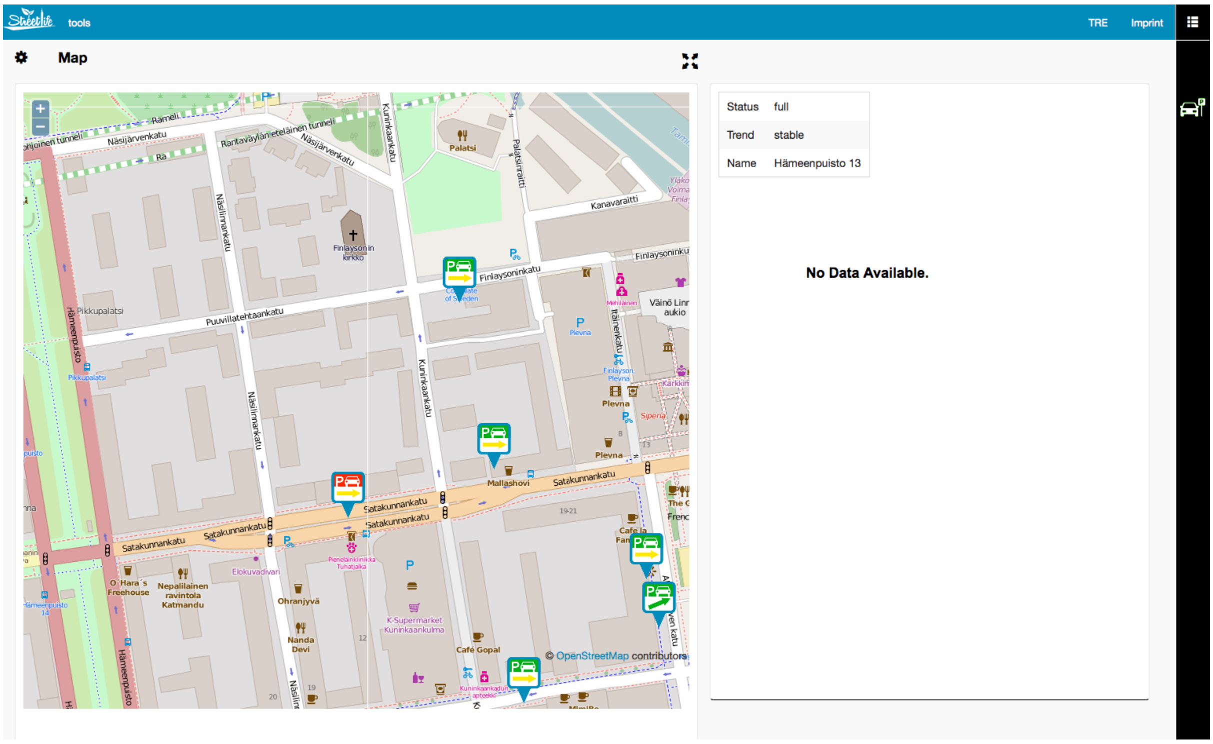

Figure 13: Tampere Park & Ride scenario

Each icon on the map describes the status of a parking facility. The status has different meanings. The background colour indicates a current situation on the parking facility and an arrow indicates a trend for a parking facility.

- A green background means that there are parking lots available.
- A red background means that there are no parking lots available.
- A green arrow means that cars are leaving the parking facility and more parking lots are increasing.
- A red arrow means that more cars arrive on the parking facility and the free parking lots are decreasing.
- A yellow arrow means a stabilised situation.

Compared to Rovereto, the mobility manager of Tampere does not use build-in functions of the MMECP to take actions. Instead she uses a system that is part of the management centre of Tampere. Another difference to Rovereto is the real-time ability of this scenario. The data in Tampere gets updated more frequently than in Rovereto. Each change of the parking lot situation will update the MMECP UI within seconds. The UI gets real time date via Web Sockets from the MMECP backend.

3.4. Bike Sharing

In the bike sharing scenario the control panel function is to provide useful instruments for the evaluation of the bike sharing system to the mobility manager. It is in the moment implemented and in action for the field tests in in Rovereto and in Berlin and used the same way. Its analysis instruments will enable the mobility manager to analyse the efficiency of the bike sharing system.

The control panel will be equipped with main features for the bike-sharing scenario. These are: the visualization of the city Open Street Map, that will show in real time the situation of the system on the offer side, displaying for each bike-station the following info:

- address of the bike sharing stations
- number of total bike slots in each station
- number of available bikes for each station
- number of free bike slots for each station

These features are also available in the ROVBikeSharing app, and are also already available in the website of the company who is in charge of the service, called Bicincittà, see [9].

Through the acquisition of bike sharing service management data, the mobility manager will be able to obtain all the information needed regarding the service given by the municipality.

The data described above will be used in the control panel to produce a summary report showing attributes and performance of the service, with the following data:

- number of bike sharing stations
- number of bikes
- number of available bikes on real time
- number of the total bikes taken from stations
- average time use of a bike
- estimate of kilometres travelled with bike sharing system
- amount of saved CO_2 consequently to the choice of a sustainable mode of transport

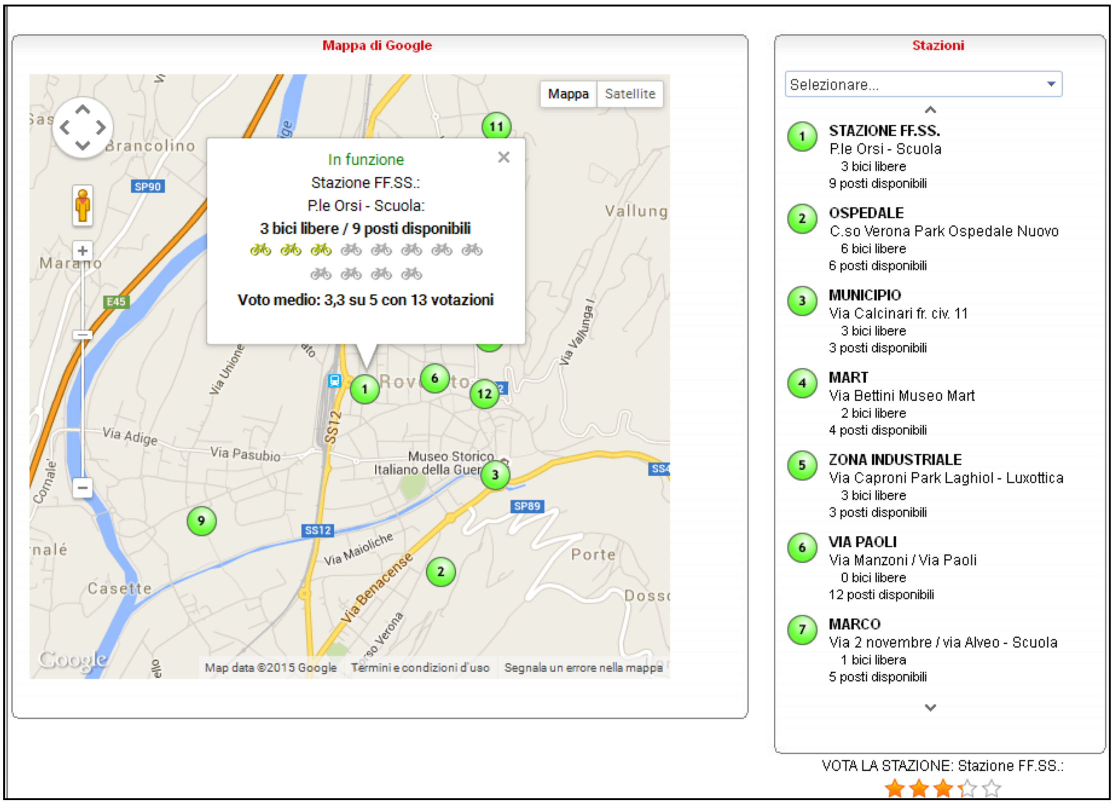

Figure 14: Example of the bike sharing offer side features, see [9]

The method used to obtain bike sharing data, in particular the monitoring of the bike identification number, will enable the control panel to identify the frequency level of trips between different stations, and indicates the main routes used by the cyclists to travel inside the city.

Furthermore, the dashboard displays the state of each bike sharing station related to the number of total implemented operations in order to have a clearer perception of the contribution given by each station, information that will be useful for the mobility manager to determine which stations are in use and which are not.

The information about the identification number of starting and arrival bike sharing station is needed to obtain the kilometres travelled in each trip. This data serves as base to estimate the measure of CO_2 saved thanks to the use of bike sharing system (despite of other means of transport less sustainable as car.

The main future development in Y3 for bike sharing scenario will be the implementation of a feature that supports the mobility manager to identify the optimal location for the stations and will give her advices regarding the size of the stations, depending on the O/D matrix calculated from historical data.

With respect to Y1, the future horizon for the bike sharing functionality inside the control panel changed in Y2: we saw it could have higher potential if it includes different types of bikes – this was done in the Rovereto pilot, since the MMECP could have represented a reliable instrument of harmonization amongst the existing bike sharing alternatives. The bike sharing feature in the MMECP in Y1 is only related to one type of service – in this case it wouldn't have many commercial possible openings, for the following reasons:

1. End users have access to real time data from the app
2. Mobility managers can check bike availability in real time also without the control panel
3. Mobility manager already has the instruments (a software provided from the company in charge of the bike sharing service) needed to compile stats and collect historic data

The main reasons behind the implementation of the Bike Sharing scenario in MMECP are not related to future commercial exploitation, but are more about the advantages and utility that the mobility manager can gain from an instrument that will enable her to consult data related to every different means of sustainable mobility on the same device.

3.5. Accidental hot spots for cyclists

To change the modal split towards carbon friendly mobility means these have to be made more attractive, they need to be: safer, cleaner, convenient: Especially cycling due to motorized traffic should to get more attractive to bring about a behavior change of the traffic participants.

This is the reason why the new STREETLIFE scenario displayed also in the MMECP for the Berlin pilot focusses on accident hot spots for cyclists.

The VMZ Berlin provides us with official accidents hot spots with the involvement of cyclists, see Figure 15. We will use this information in order to visualise it as a layer on the map of the MMECP.

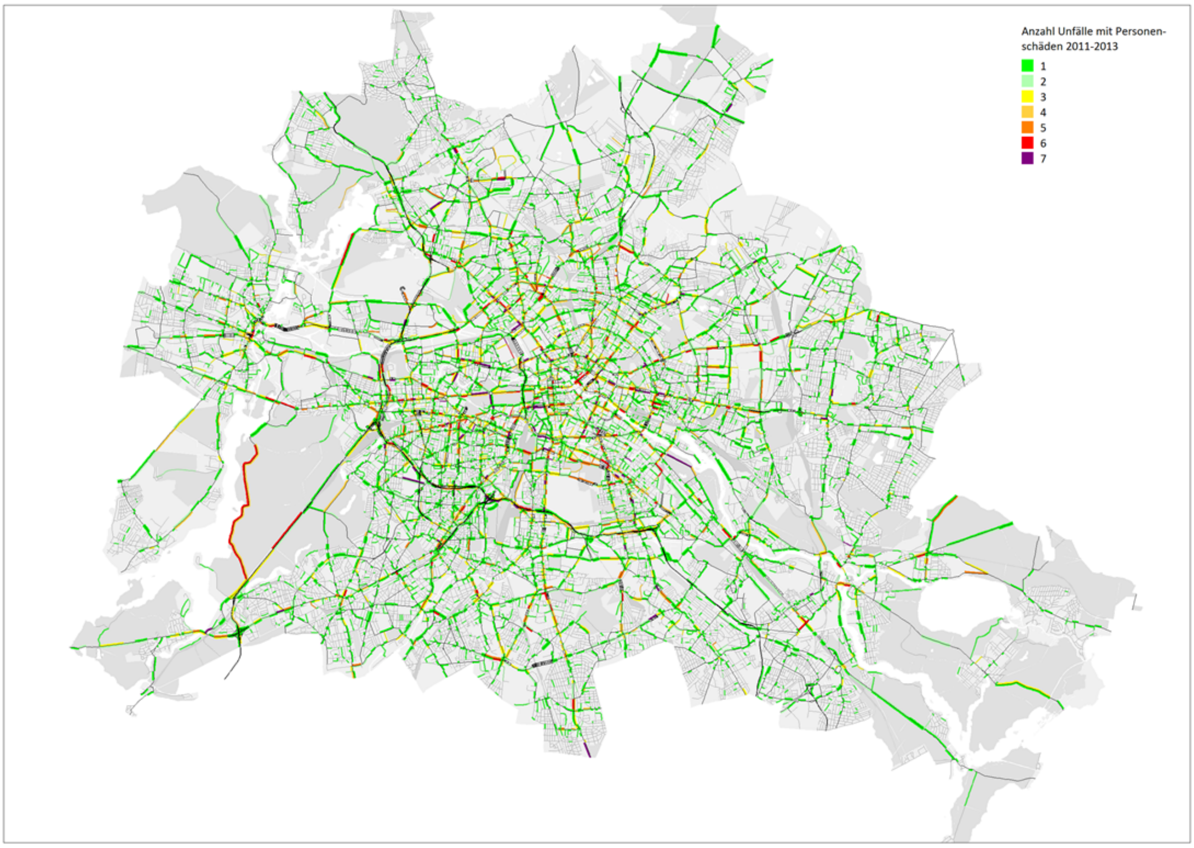

Figure 15: Accidents hot spots of Berlin 2011 to 2013

In October 2015 a new Berlin pilot test starts. Within this test, users are asked to use the Berlin STREETLIFE routing app for their daily trips and to track their movements. After each trip, the app asks the user to give feed-back on his perception of the level of safety during this specific bike route. The result is an origin destination matrix of each trip of the users. This includes the information of the modal choice of the users. The MMECP will visualise that information as a second layer on the map.

The MMECP mobility manager can now compare official safety data of the city with subjective values from the STREETLIFE users. This is very valuable data and can be used for instance in order to investigate the impact of former construction sites and the usefulness of the provided routes of the safety routing service of the VMZ.

4. DEVELOPMENT IN A GENERIC INTERFACE

Another technical development that took place in Y2 within the MMECP was the release of an extended version of the generic interface specification the standardized retrieval and evaluation of spatial-temporal data in STREETLIFE.

The interface defined in the following sections is a revised and extended version of the specification provided in D4.2.1, see [6]. It includes the possibility to provide the client with metadata, which is important information for the proper handling and visualization of the actual raw data.

The interface provides a generic way to select, structure, aggregate, and retrieve spatially and/or temporally localized measurement data in a standardized manner. It is generic in the sense that it is neither specific to a particular type of database, nor is it specific to a particular programming language. The implementation in a particular programming language should be straightforward. A key feature of the interface design is that all the data can be arbitrarily distributed in space and time. It does not require that the data is arranged regularly in any way. In particular, the measurements of the quantities are not required to be synchronous or equally spaced. This said, for best practical use it is nonetheless often beneficial if the measurements were performed frequently and on a regular basis.

The extended interface proposed here consists of only two functions. They can be implemented, for example, as remote procedure calls (RPC) from a client to a server hosting the data (e.g., by using HTTP-based calls):

1. an initialization function,

   (*list_of_functions, list_of_measurement_quantities, list_of_condition_keywords*) = **get_keys**(),

2. The actual retrieval function,

   array_of_scalars = **get_data**(
   list_of_functions,
   list_of_measurement_identifiers,
   list_of_conditions0,
   list_of_conditions1,
   list_of_conditions2).

The types of objects required are arrays of floating point numbers, associative arrays, and (ordered) lists. Theoretically, the minimum requirement for this interface is that each list contains just a single element. In this case, the resulting *array_of_scalars* would contain just a single floating point number. However, for a more efficient use of the interface, we will consider multi-element lists and four-dimensional arrays. As list elements we mainly use text strings, which have the benefit that their meaning can be largely self-explanatory.

The initialization function **get_keys** receives no parameters and returns three lists specifying the capabilities of the server with respect to the retrieval function **get_data**. They effectively define the vocabulary understood by the server. In trusted computing environments one might consider adding a *clientID* input parameter to **get_keys** in order to provide different clients with different sets of capabilities. In untrusted environments one should rather resort to more reliable authentication methods though.

4.1.1. List of measurement quantities

The *list_of_measurement_quantities* returned by **get_keys** consists of a list of associative arrays. Each array in that list contains the metadata of a particular scalar measurement quantity that can be queried in the database of the server. The metadata is given in terms of a number of (key, value)-pairs and specific text strings are used as keys. A mandatory (key, value)-pair is the (“identifier”, *measurement_identifier*)-pair. The *measurement_identifier* can be a number or a string that uniquely identifies an accessible measurement quantity in the database comprising a multitude of scalar measurements in space and time. Each individual measurement should be associated with a timestamp and a spatial location (e.g., latitude longitude, and elevation). In case of missing timestamps, the affected measurements should be ignored when temporal constraints are specified in a query. In case of missing location information, the affected measurements should be ignored when spatial constraints are given. In other words, constraints are considered unfulfilled if the respective information is missing.

To give an example, a *measurement_identifier* could be a text string like this:

"BikeSharingCompanies(FlexiBike).NumberOfAvailableBikes"

This identifier would be associated with measurements of the number of available bikes in all bike sharing stations of a particular bike sharing company.

The *list_of_measurement_identifiers* provided as a parameter to **get_data** is a (non-empty) list of these *measurement_identifiers*. A corresponding number of results with regard to the respective measurement quantities will be returned in the corresponding order, as shown in detail further below.

Besides this essential (key, value)-pair, other metadata information about a measurement quantity might be required, e.g.

- "locations" (key) – value: a list of all existing measurement locations for the given quantity. Each element in the list may contain the specific coordinates and/or a string with the name of the location. This allows the client to query data from specific locations (in **get_data**). In the bike sharing example, these could be locations of particular bike sharing stations.

- "areas" – a list of spatial areas where the data is measured. An area can be specified, for example, as a polygon on a two-dimensional surface, and/or by a name given in a string. This provides the client with another possibility to filter data spatially.

- "measured since" – timestamp of the first available measurement. Such a timestamp can be specified as appropriate, typically as RFC 3339 or ISO 8601 timestamp.

- "measured until" – timestamp of the last available measurement. A special indicator timestamp can be defined to designate ongoing measurements.

- "name" – a string containing the (human-readable) name of the measured quantity, e.g., "number of available bikes".

- "unit" – a string containing the unit of the measured quantity, e.g., an SI unit like "kg" or "m". Measurement quantities without unit (e.g. "number of available bikes") can be denoted with an empty string.

  Categorical quantities: A special case can be introduced to enable the processing of measurements of categorical variables with this interface. For this purpose, the value of "unit" can be given as a list of strings, each string denoting a category, e.g., "red", "green", and "blue". The measurement of a categorical variable can then be represented in terms of scalar numbers (hence fitting into the given framework), each number constituting the ordinal number of a category in the list of category names. Note, however, that it generally does not make sense to use aggregation functions like "mean" and "median" in conjunction with ordinal numbers. Therefore, the access to categorical data is mainly confined to the use of the function "n" in combination with the filter condition "value_is" (see below).

- "description" – a string that contains a textual description of the measured quantity.

Other (key, value)-pairs of metadata can be added, if necessary.

4.1.2. Functions, conditions, keywords and scalars

The specification of the *list_of_functions, list_of_condition_keywords,* the *list_of_conditions,* and the *array_of_scalars* did not change with respect to D4.2.1 except for a name change: the *list_of_attribute_identifiers* in D4.2.1 is now called *list_of_measurement_identifiers*. Likewise, an "attribute" in D4.2.1 is now called "measurement quantity." We refer to D4.2.1 for further information on the definition of these lists.

4.1.3. Conclusion

We presented a generic interface specification for standardized retrieval and statistical evaluation of spatial and temporal measurement data. It is not specific to a particular type of database, programming language, or data format. It also does not demand a specific syntax nor is the scope of its functionality completely fixed. In this way, it allows for an easy adaptation to the specific needs and requirements of a particular software implementation. The generic interface is already successfully used in the MMECP for gathering data for the Tampere scenario. A paper ("A Generic Interface Specification for Standardized Retrieval and Statistical Evaluation of Spatial and Temporal Data") on the presented interface specification written by participants of WP4 was accepted for publication and presentation at the 7th International Conference on Knowledge Management and Information Sharing 2015 (KMIS 2015).

5. DEVELOPMENTS IN SIMULATION

In Y2, the two planned simulation functions within the MMECP made technical progress. These two are the so-called "micro simulation for Berlin" and the "macro simulation for Rovereto". This section describes the progress that has been made.

5.1. Berlin

The BER simulation scenario covers an area of about 75 square kilometres. The centre of this area is the former Tempelhof Airport, which is now known as Tempelhofer Freiheit and is one of Berlin's event locations. With its two take-off and landing strips, the emerging Tempelhof Park offers over 300 hectares of open space for a number of events, including sport tournaments, open-air concerts and art and educational projects. The BER simulation scenario is to model the situation n before and after a big event at Tempelhofer Freiheit. A huge number of people arrives at the event area before the event starts and is willing to go home after the event. Several people will use a bicycle for their trip. Since a huge number of other bicycles, vehicles, and pedestrians are on their way in the considered area, many dangerous situations can occur for bicyclists. To reduce the risks to bicyclists, the bicycle router of the VMZ is to be used to suggest each bicyclist safe routes that circumnavigate dangerous road segments and intersections. One aim of the BER simulation scenario is to detect which safety benefit can be achieved for the bicyclists by using the VMZ bicycle router.

The simulations start is planned for September 2015. In the last months, the simulation architecture VSimRTI was extended and data converter and new interface were implemented for fulfilling the special needs of the BER simulation scenario. Furthermore, the BER simulation scenario was prepared.

Network and Hazard Spots

For the modelling of the bike paths, the official Berlin detail network is used. The network covers all official streets and in Berlin. The data of this network is provided by the VMZ in the shapefile format. Converter were developed to import the data into the database of the simulation architecture VSimRTI. This simulation architecture is used to model the individual movement and behaviour of all bicyclists from their starting points to their destinations in the BER simulation scenario. Figure 16 shows an extract of the network.

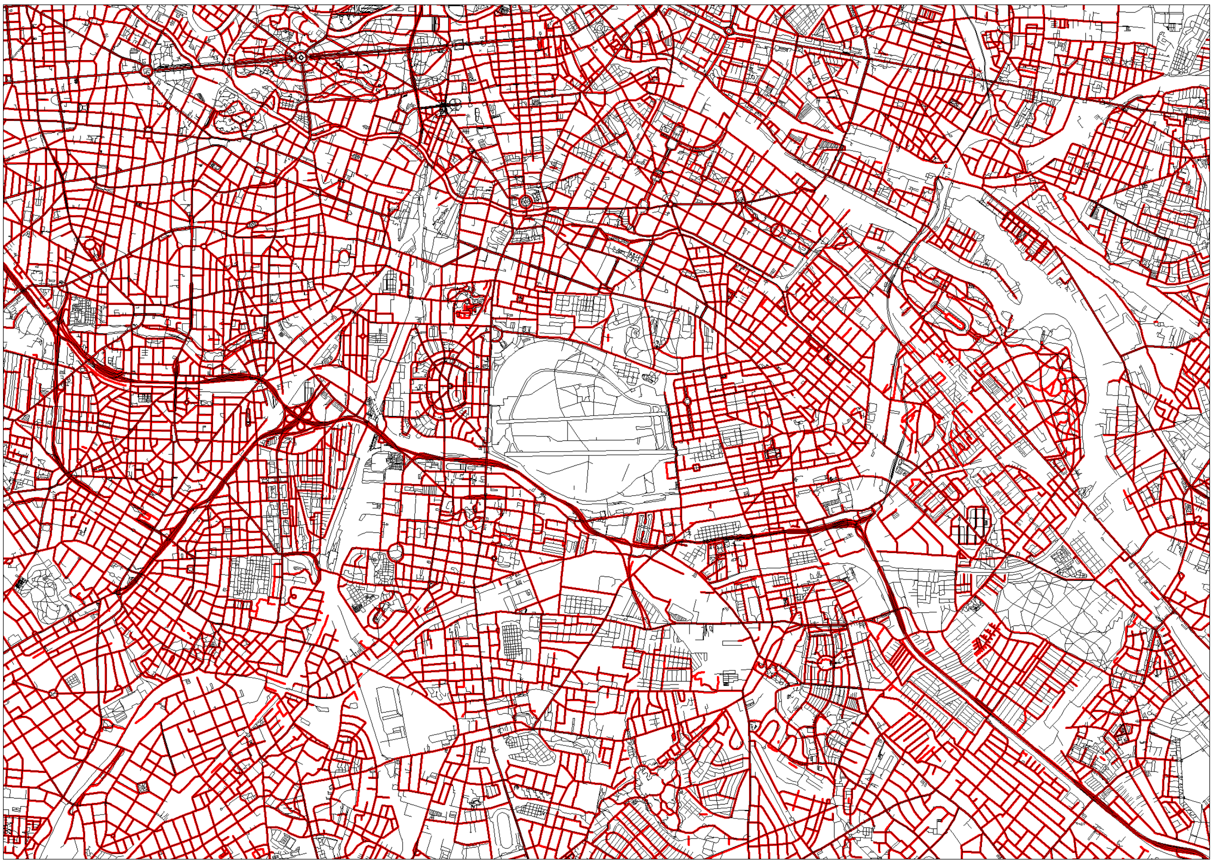

Figure 16: Network of the city centre of Berlin (Source: VMZ)

To detect which safety benefits can be achieved for the bicyclists by using the VMZ-bicycle router, information about hazard spots are needed. For this reason, information about hazard spots of the city centre of berlin are provided by the VMZ. This data is available in the shapefile format and converter were developed to import the data into the database of the simulation architecture VSimRTI. The following information about hazard spots is planned to be used for the evaluation of the simulation results:

- Number of accidents with damage to persons on a hazard spot (years 2011, 2012, and 2013)
- Number of accidents with involved bicyclists and with damage to persons on a hazard spot (year 2013)
- Number of accidents with damage to persons on a hazard spot on a road lane (year 2013). Accidents on bike lanes and pedestrian paths are not included.
- Number of accidents with strong damage to persons on a hazard spot (years 2011, 2012, and 2013)
- Number of accidents with involved bicyclists and with strong damage to persons on a hazard spot (year 2013)

Bicycle Routes

In addition to the network and hazard spots, Origin-Destination (OD) Matrices are needed to specify the travel demands between origin and destination nodes in the traffic network. With this information, a realistic route calculation for all bicyclists is possible and, then, the movement of the bicyclists can be modelled by the simulation framework VSimRTI. For the BER simulation scenario, the Origin-Destination Matrices were generated by using the microscopic travel demand model TAPAS (Travel Activity Patterns Simulation) of the DLR. As a result, the origin and destination points of 5,389 bicyclists, which rode a bicycle to the big event at Tempelhofer Freiheit and back, could be calculated.

Figure 17 shows the starting points of the bicyclists riding to the big event at Tempelhofer Freiheit.

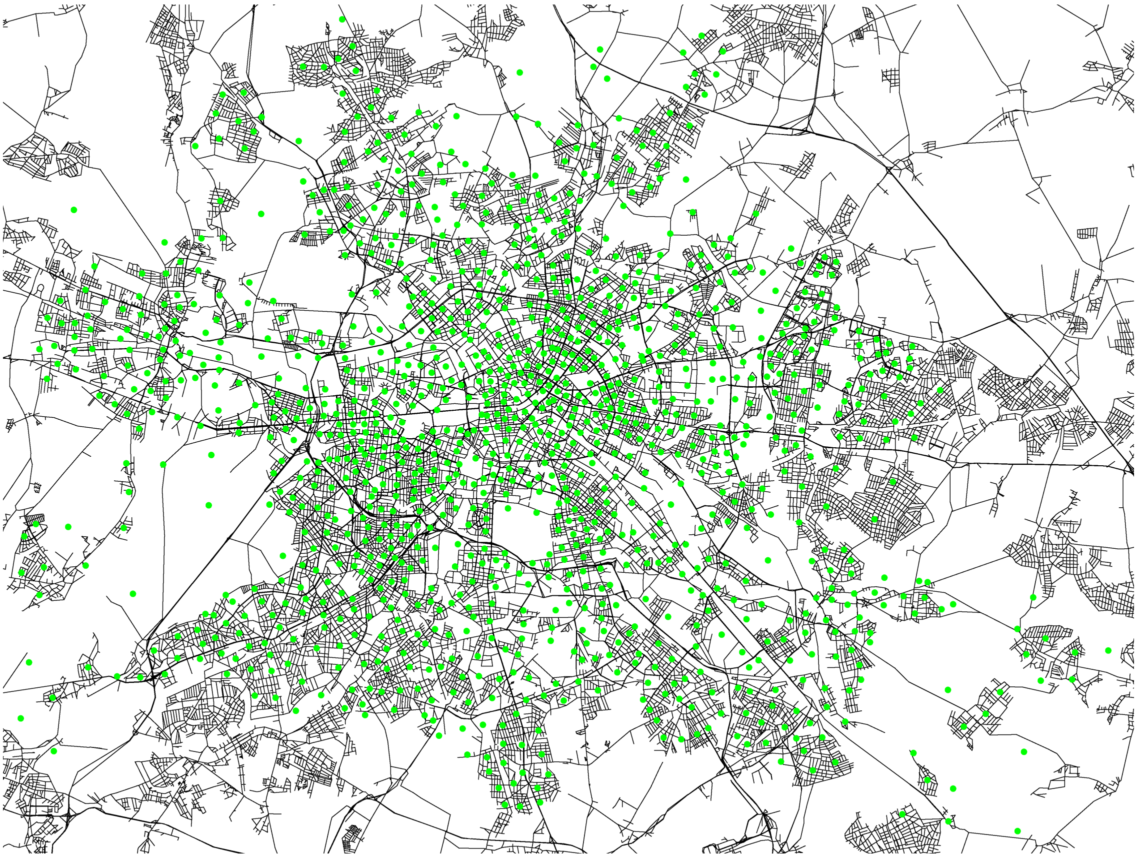

Figure 17: Starting points of the bicyclists riding to the big event at Tempelhofer Freiheit

5.2. Rovereto

The simulation scenario in Rovereto aims at testing the MMECP-functionality as a helpful instrument for a mobility manager that has to decide the most useful premises for a new parking facility in town, the optimal size of these facilities, and how to incorporate these new service into the existing traffic system with multimodal options.

One of the requirements for an optimal use of the Park & Ride simulation functionality is the level of knowledge basis that belongs to the mobility manager: she needs to know in advance where the most optimal places for a new parking facility are, in order to reduce the number of possible simulations and use the instruments in a proper way. In the Rovereto pilot, the existing urban mobility plan has already established the best available spots for new parking facilities (marked in red in the below city parking map), so the mobility manager will be enabled to use MMECP in order to test location, size and mobility services.

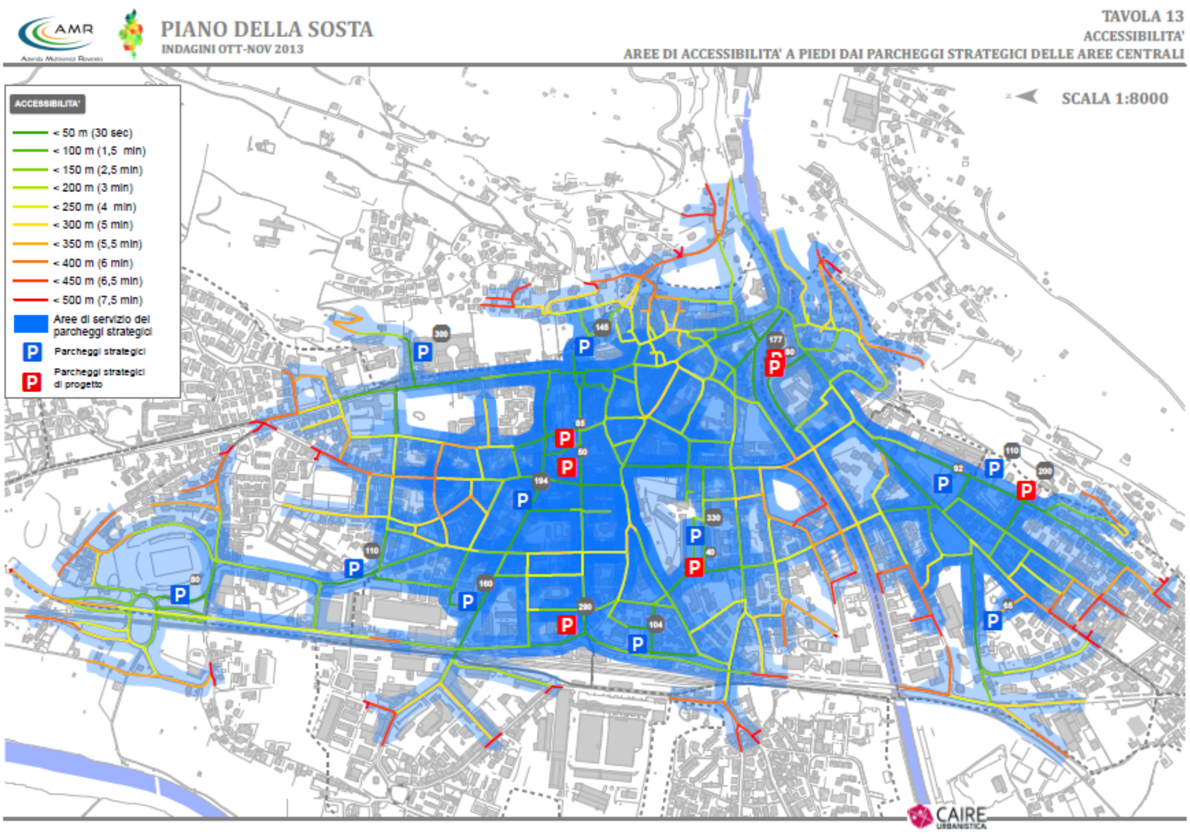

Figure 18: Location of future parking facilities (source: Rovereto Parking Plan)

Starting the simulation scenario requires in advance the definition of a set of features regarding the new parking lot which the mobility manager needs to define. The information that needs to be provided by the mobility manager and given as an external input to the MMECP for the simulation of a new Park & Ride area are the following ones:

- location of the new Park & Ride area
- number of parking slots calculated on the basis of the surface of the area

Once those data are established, the manager will have all the information needed by the simulation model in order to calculate the impact of a new Park & Ride area.

The starting point of the simulation scenario for the Rovereto pilot is that the mobility manager opens first the simulation feature that enables her to input in the MMECP the main features of the new parking lot: number of slots, different ride services (e.g. a bike station with 6 bikes, or a bus stop with a vehicle every 30 minutes). When these parameters will be given to the system, the mobility manager will see the simulation already gathered simulation results to his specific parameters.

The simulations will be run offline by using a traffic model that has been developed for the urban mobility plan of the city of Rovereto with the aid of Cube 6.0 software [10] so as to analyse phenomena related to mobility system in the town territory.

The model contains an Origin-Destination matrix with journeys carried out with cars and heavy vehicles. This matrix has been built with estimate techniques based on socio-economic data and results from ad-hoc surveys. During the simulation process the matrix is subject to adjustments in order to represent with an adequate accuracy the state of the art.

The simulation of a new parking area will follow these steps in the traffic model:

Identification of which OD relations are interested by the new parking lot using a Cube feature that allows to determine the number of vehicles passing on a specific road with a particular origin and destination. The identification of the area and the type of parameters chosen are the trigger needed in order to select the optimal simulation for the selected scenario between the ones provided. The simulations run off-line, then the control panel will select the scenario that matches with the selected variables.

After the simulation phase is over, the Mobility Manager will receive two different outputs from the simulation that will enhance her knowledge base and will help her take decisions.

The first output is a report with the expected impact on the traffic system derived from the new parking facility: This report will show in first place the parameters that the Mobility Managers has chosen, like the number of projected slots and the type of ride services activated (public transport, bike sharing stations, etc...) in order to get a resume of the selected scenario. The results of the simulation shown in the report will be the new estimated occupancy rate, the number of kilometres not travelled by car in the scenario, the CO_2 saved thanks to the insertion of the new facility in the system.

The second output won't be shown on the report, but directly on the city map: new occupancy rates for existing parking lots depending on the impact of the new one, new occupancy rates also for macro zones of the city, position of the new parking facility, and the traffic-assigned model of the road network. The other result of the simulation scenario will be the update of access cost indicator in the macro zones interested by the new facility.

A future development (after the project) could be a use case where different simulation scenarios could be compared on the screen.

6. Conclusion

In this deliverable we explained where the MMECP has extended since STREETLIFE Y1: In Y2 the MMECP UI was improved, most scenarios were implemented, the common API got an important extension and the implementation for the simulations are ready to deliver results.

Due to limitations of the GWT, the UI was re-implemented with AngularJS and shown in Section 2. So far, the new MMECP UI proved its usefulness and is now ready for the integration of the scenarios, which are the “Forecasting mechanism for Berlin”, the “Bike Sharing Scenario for Rovereto and Berlin” and the New Berlin scenario showing accident hot spots for cyclists. Section 3 elaborates on the scenarios for Berlin, Rovereto and Tampere. The Forecasting Scenario in section 3.1 describes two mechanisms, especially the bike availability forecast results in an API that is already deployed at CIP site and will be used in Y3. In Section 3.2 we discuss the three approaches distinguished to best estimate CO_2-Emission in Berlin in the moment. Due to the lack of data, it is not possible to provide an accurate estimation of CO_2-Emission, but we can show hotspots of CO_2-Emission, which should need further investigation by the city administrates. Section 3.3 describes the Park & Ride Scenario for Rovereto and Tampere and explains their differences, especially in case of the real time capabilities. The bike sharing scenario (section 3.4) will be used for Rovereto and Berlin in the same way. As the newest scenario, we presented the accident hotspots for cyclists in section 3.5. That scenario shows a valuable usage of the feedback data that will be collected by the Berlin pilot test in October 2015. Section 4 shows the updated common API, especially with extensions to spatial and temporal measurements. In addition to the updates of the API we can say that our paper "A Generic Interface Specification for Standardized Retrieval and Statistical Evaluation of Spatial and Temporal Data" was accepted by KMIS 2015. As the acceptance of the paper are recent news, it is not listed as a reference yet. Finally, Section 5 shows the status of the Simulation-functions for Berlin and Rovereto. The Berlin simulation is ready end of September, in parallel with the completion of the implementations of the VMZ cycling router. The simulation demonstrates if the suggested route is safe for cyclists. Rovereto will use its simulation in order to support the mobility manager to improve the Park & Ride system performance.

As next steps WP4 will finish the implementations of the MMECP and its simulations and prepare the pilot tests at the beginning of Y3. The results will be used to improve the system within Y3.

APPENDIX A: LITERATURE

[1] "GWT", 30-Sep-2015. [Online]. Available: http://www.gwtproject.org/

[2] "AngularJS - Superheroic JavaScript MVW Framework", 30-Sep-2015. [Online]. Available: https://angularjs.org/

[3] "JavaServer Faces Technology", 30-Sep-2015. [Online]. Available: http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html

[4] "Apache Wicket", 30-Sep-2015. [Online]. Available: http://wicket.apache.org/

[5] "Sencha Ext JS", 30-Sep-2015. [Online]. Available: https://www.sencha.com/products/extjs/#overview

[6] B. Dittwald, S. Cuno, J. Kohlmorgen, K. van Bernum, B. Schünemann, B. Obst, V. Brandstetter, R. Kelpin, D. Frigeri, N. Perri, F. Avesani, A. Merigo, C. Schwarz, "D4.2.1 – Mechanisms and tools for mobility management and emission control (initial)" Sep-2014

[7] Dr.-Ing. Frank Ließke, R. Dr.-Ing Wittwer, S. Dipl.-Ing. Hubrich, and G.-A. Prof. Dr.-Ing. Ahrens, "Endbericht zur Verkehrserhebung 'Mobilität in Städten - SrV 2008' in Berlin," Aug. 2009. (Google 2015)

[8] M. Mitrevska, G. Valetto, M. Vuorio, and A. Nurminen, "D3.2.1 – Mobility Data Integration Models and Techniques (initial)." Oct-2014.

[9] " Bicincittà", 30-Sep-2015. [Online]. Available: http://www.bicincitta.com/

[10] "Cube", 30-Sep-2015. [Online]. Available: http://www.citilabs.com/software/cube/