Project number:	288577
Project acronym:	urbanAPI
Project title:	Interactive Analysis, Simulation and Visualisation Tools for Urban Agile Policy
Instrument: **Call identifier:** **Activity code:**	STREP FP7-ICT-2011-7 ICT-2011.5.6 ICT Solutions for governance and policy modelling

Start date of Project:	2011-09-01
Duration:	36 months

Deliverable reference number and title (as in Annex 1):	**D3.8 Simulation Components Documentation**
Due date of deliverable (as in Annex 1):	**31.8.2013**
Actual submission date: **Revision:**	*see "History" Table below* **1**

Organisation name of lead contractor for this deliverable:
Fraunhofer IGD

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2013)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium	
CO	Confidential, only for members of the consortium (including the Commission Services)	

European Commission
Information Society and Media

Title:
Simulation Components Documentation
Author(s)/Organisation(s):
Jan Peters-Anders / AIT, Ernst Gebetsroither / AIT, Wolfgang Loibl / AIT, Jens Dambruch / Fraunhofer IGD
Working Group:
WP3 – Fraunhofer IGD, AIT
References:
Description of Work

Short Description:
The report describes the implemented simulation components of the three UrbanAPI applications in development. It covers the server and client side developments done so far.
Keywords:
Simulation components and functionalities, graphical user environments, operation, graphical user interfaces

History:

Version	Author(s)	Status	Comment	Date
001	Wolfgang Loibl	new	Initial draft table of contents, initial content	12. 07. 2013
002	Ernst Gebetsroither	rfc	Added UGS to chapter 2	
003	Jan Peters-Anders	rfc	Added Public Motion Explorer to chapter 2	29.08.2013
004	Wolfgang Loibl	rfc	Minor corrections and extensions	30.08.2013
005	Jens Dambruch	rfc	Added remarks on 3DVR	01.09.2013
006	Ernst Gebetsroither	final	Rework after review	16.09.2013

Review:			
Version	Reviewer	Comment	Date
005	Wolfgang Stemberger Venco Bojilov	Minor changes and corrections	8.9.2013

About this Document

This document is deliverable D3.8 of work package 3 of the urbanAPI project, due in PM24. This document deals with the simulation tasks. This task encompasses the development of server- and client-side simulation components for all three applications. The report describes the implemented simulation components.

Role of Partners:
Task 3.1 AIT develops simulation components, and supports the application partners in defining their application using the toolset.

The document will be superseded by an improved version if required at any time in the project life cycle. The finalized version of this document is planned for month 30.

1 Introduction

A model is an abstract representation of reality for a certain purpose carried out by building a system of the major elements and relations describing a certain issue for a certain purpose. Simulation is the process of modelling the dynamics of a system and its particular elements in space and time.
Modelling and simulation (M&S) allows getting information about how a system behaves without actually testing it in real life.

According to the DOW, the following work is to be done:

- To enhance the model structure by increasing agent classes, behaviour rules and control functionality and to define the agent classes and behaviour rules based on analysis of social data exploring the prior urban socio-demographic spatio-temporal dynamics.
- To create grid cell layers describing land use state, attractiveness, repulsion and suitability for growth or densification based on the land use, the distance to other land uses and the fraction of certain characteristics within an influence area.
- To program enhanced agents‘ actions, searching appropriate living space or business areas. Fraunhofer IGD will support this task by implementing simulations in GPU languages, so that they can be run in real time. This task is not included in this version of the document as it is still under research.

2 Simulation components by applications

2.1 Application 1 – 3DVR

In the current version there are no special simulation components for the 3DVR application. However, all things dealing with 3D graphics, which can also be considered as a kind of simulation, are covered in other deliverables, mainly 3.2 and 3.6.

2.2 Application 2 – Public Motion Explorer

The UrbanAPI Public Motion Explorer (PME) is designed as web application that is capable of visualising mobile user distribution patterns of pre-processed mobile device data. It has been developed from scratch, i.e. from a generic parser for the binary GSM log data to the web interface of the application. Its simulation capability lies in the possibility to choose a so called source (a cell where users are originating from) or target (a cell where users go to) and to simulate their motion over the day. The following describes the application that has been developed for the city of Vienna because the Austrian provider's data has the by far best spatial and temporal resolution.

2.2.1 Components and functionalities

The main PME simulation components are –at the client side- the web based graphical user interface, including user controls and a map frame to visualise the results of the user request (for a description of the user interface see also D3.6 *Beta Applications 1-3*) and on the server side the PostgreSQL/PostGIS database holding on the one hand the data to process and on the other hand the functions to do the processing. The PME simulation process can be schematically depicted as in Figure 2-1.

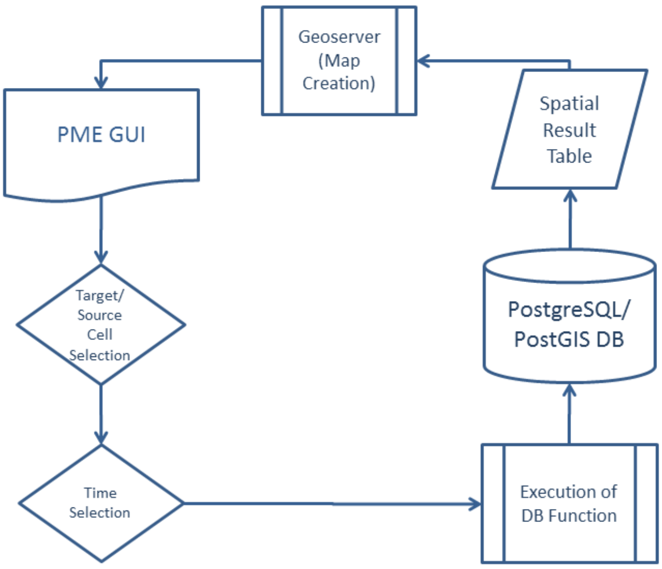


Figure 2-1: Schematic data base request of the PME application

The database of the PME consists of a pre-processed dataset derived from raw log data of the different mobile providers' data used in the project. It has the following layout and data types (Table 2-1):

src_cell_id_0_to_4	timestamp	tgt_cell_id	count	the_geom	pk
10kme479n280	2012-01-24 00:45:00	1kmn2806e4792	1494	0106000020DB0B000(	703731
10kme479n280	2012-01-24 01:30:00	1kmn2806e4792	1493	0106000020DB0B000(	684340
10kme479n280	2012-01-24 01:15:00	1kmn2806e4792	1493	0106000020DB0B000(	707419
10kme479n280	2012-01-24 00:30:00	1kmn2806e4792	1493	0106000020DB0B000(	680465
10kme479n280	2012-01-24 01:45:00	1kmn2806e4792	1493	0106000020DB0B000(	666401
10kme479n280	2012-01-24 01:00:00	1kmn2806e4792	1492	0106000020DB0B000(	716805
10kme479n280	2012-01-24 00:15:00	1kmn2806e4792	1492	0106000020DB0B000(	669420
10kme479n280	2012-01-24 02:30:00	1kmn2806e4792	1490	0106000020DB0B000(	679520

Table 2-1: PME mobile user main dataset layout

The data types used for the 6 columns have the following data types (Table 2-2):

Name	Type	Length	Decimal:	Allow Null	
src_cell_id_0_to_4	varchar	0	0	☑	
timestamp	timestamp	6	0	☑	
tgt_cell_id	varchar	0	0	☑	
count	int4	32	0	☑	
the_geom	Type	0	0	☑	
pk	int4	32	0	☐	1

In order to enable the user to select a certain region he or she is interested in, the map of the PME contains a selection layer which is derived from the official Statistics Austria 1 and 10 km ETRS89/LAES grids (to be found here: https://www.statistik.at/web_de/services/geodaten/index.html) (Figure 2-2).

Outside Vienna where the population distribution is less dense, 10x10km raster cells are defined as target or destination cells. Inside Vienna 1x1km raster cells are defined as target or destination cells. All interaction data have been aggregated this way in advance and stored in a database. The controls in the user interface are triggering SQL queries to extract the subset of a certain mobile phone user group - e.g. those of an origin cell approaching destination cells within certain time steps or – the other way round – a certain mobile phone user group occupying a destination to show the origins of these cell occupiers.

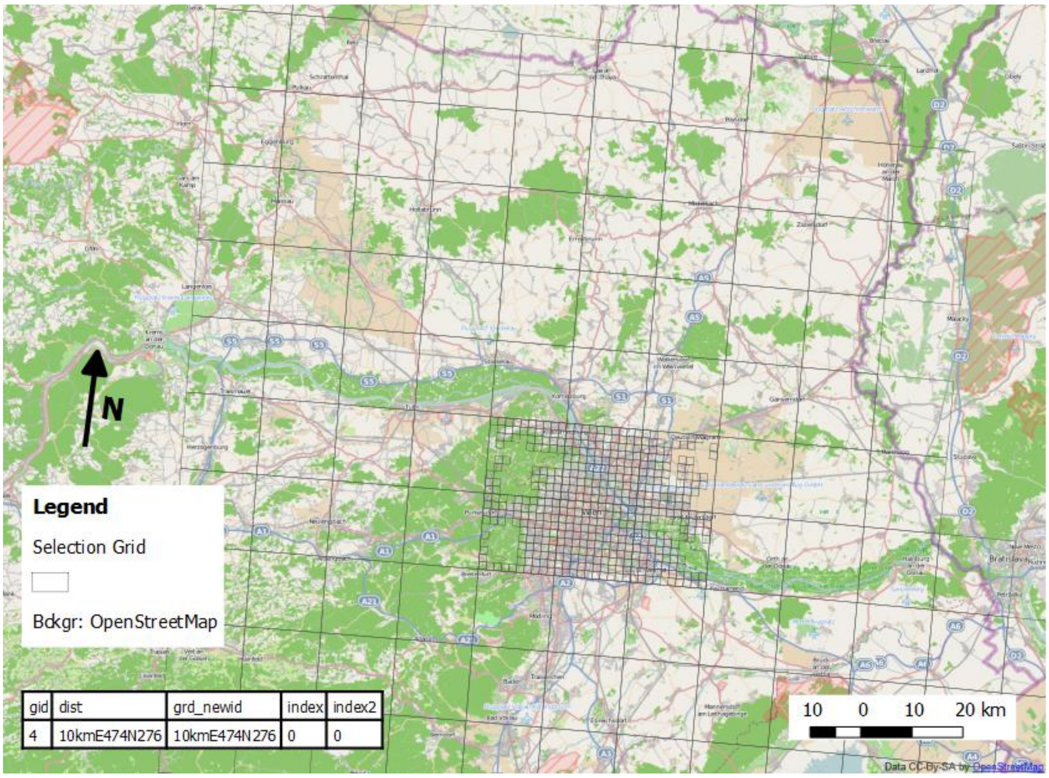


Figure 2-2: PME selection grid derived from official Statistics Austria grids for Vienna and its surroundings

These data are used to perform the following simulation tasks:

2.2.1.1 Dynamic mapping of the diurnal mobile user / population distribution

The dynamic mapping of the diurnal mobile user / population distribution is conducted via sending the user's requested grid cell id to the webserver and fulfilling the following data base SQL request:

```
INSERT INTO temp_user_selections (ts, tgt_cell_id, count, the_geom, session_id)
   SELECT timestamp,tgt_cell_id, sum(count) as count , max(the_geom)::geometry as
   the_geom, 233 as session_id
    FROM public._1_and_10_km_cell_results_0_to_4_concat
     WHERE src_cell_id_0_to_4 = '1kmn2807e4790'
      GROUP BY tgt_cell_id, timestamp;
```

Figure 2-3: Exemplary SQL command to derive a result map for the selected source grid cell

The user is then presented with a resulting heat map (Figure 2-3).

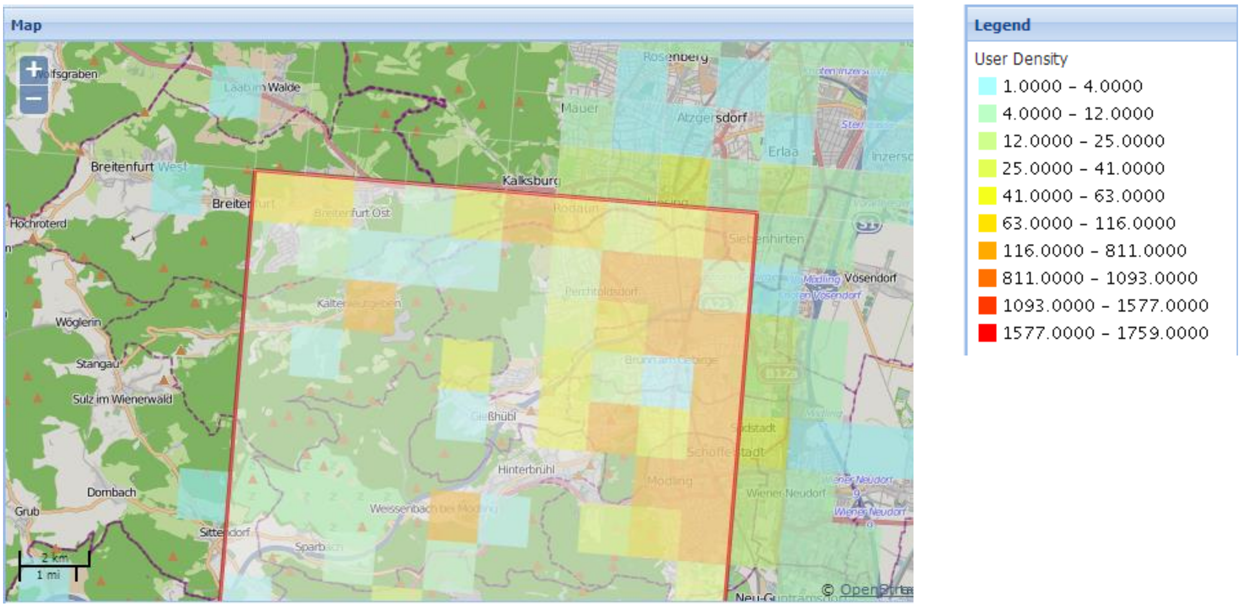


Figure 2-3: Result heatmap according to user selection (red rectangle showing the selected raster cell)

Via the above depicted process it is (at this stage of the project) possible to analyse the data set according to two regions of interest:

2.2.1.2 Dynamic mapping of the origin-based diurnal movements targeting the destination area

The first step of the motion analysis requires selecting a source region. This region is defined as the region where the depicted users are originating from (in the best case: where they are living, i.e. sleeping at night). Via the time slider the user can animate the movements of the users from the source cell to all target cells over the day and see their distribution density for each target cell for each time step on the map (Figure 2.4).

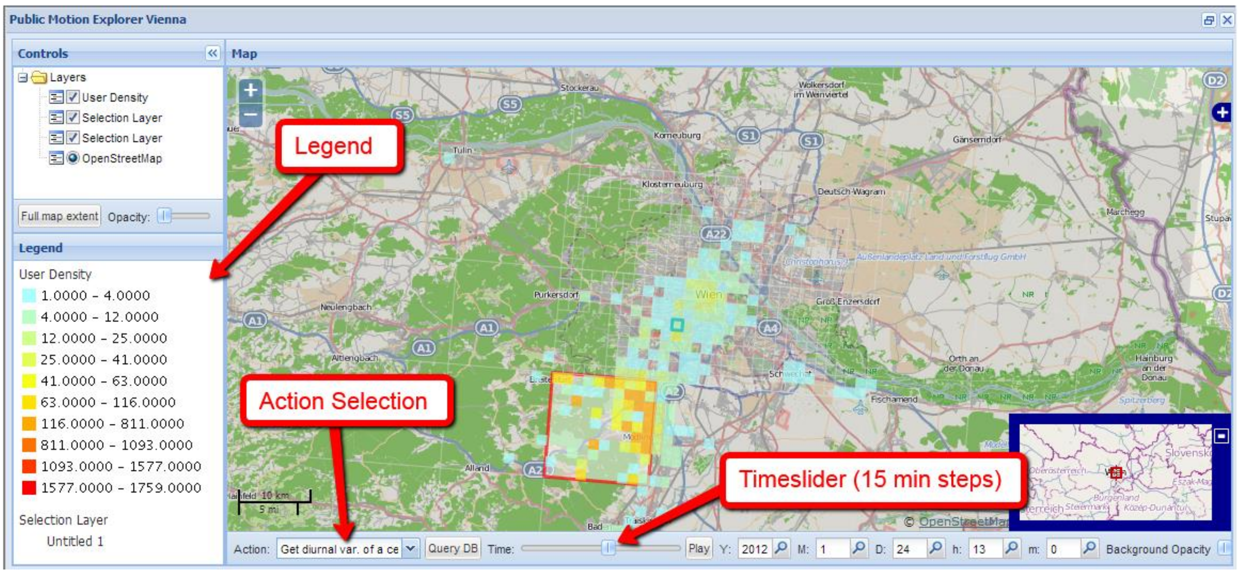


Figure 2-2: User distribution for a certain time step

2.2.1.3 Dynamic mapping of the destination-based diurnal movements showing the origin area

In the second analysis case the movements of the mobile phone users from all origins towards a selected target cell occupied at a certain time of the day is depicted. The time slider indicates the time steps, the map shows the volume per origin cell from which the mobile phone users come to move to a destination cell.

In both cases the PME tool user has the possibility to click on a resulting raster cell and see the users' distribution for this cell. This is done on the server side via the following SQL statement:

```
SELECT  ts, count
  FROM temp_user_selections
   WHERE session_id = 239 AND tgt_cell_id = '1kmn2804e4788'
ORDER BY ts
```

Figure 2-3:SQL statement to derive the diurnal variation for a selected raster cell

2.3 Application 3 – Urban Growth Simulation

The urban growth simulator (UGS) is based on the simulation suite MASGISmo – developed by AIT (Gebetsroither, 2010) combining several modelling tools within one platform as RepastJ[1] (http://repast.sourceforge.net/) for agent-based modelling,

2.3.1 MASGISmo components and main functionalities

MASGISmo combines System Dynamics (SD) modeling, Agent-based modeling (ABM), and GIS data analysis and viewing functionality. Combining SD and ABM is based on the pioneering works of Akkermans and Scholl (Akkermans 2001, Scholl 2001a, and Scholl 2001b) and Pourdehnad, Schieritz, and Milling (Pourdehnad 2002, Schieritz & Milling 2003, and Schieritz & Groessler 2003). MASGISmo combines SD modelling (through inclusion of Vensim) and ABM (through inclusion of RepastJ). Enhancing the spatial capabilities of the ABM module has enabled the inclusion of GIS data analyses within the multi-method platform. This allows users of MASGISmo to develop multi-method models simulating spatially explicit actions of agents changing land use and to spatially analyze the results of these actions. The calculation of new geographic maps out of the existing ones can be performed by using simple arithmetic operations and the agents' spatial movement by transforming land-use of single cells into steady land-use transitions. This

[1] Java Version of the Recursive Porous Agent Simulation Toolkit (Repast)

process is part of the spatial data analysis features of MASGISmo and is one main difference to other tools like Anylogic.

The development of the simulation platform MASGISmo is predominantly determined by the requirements of the projects it serves, i.e., the objects to be modeled and the modeling purposes. Almost with every model built up with MASGISmo, new functionalities for the platform are developed, serving other future modeling purposes.

The screenshot below presents the GUI of one MASGISmo simulation showing some results (Fig. 3). Three main parts characterize MASGISmo's GUI: first the general simulation controls, second the interactive toolset and third the illustration tools such as dynamic results map, GIS layer legend and the overview map. This depicted GUI is an example of the early stage of MASGISmo's development for the urbanAPI UGS. In this use case, importing GIS data of, e.g., different urban zoning plans, new infrastructure, or shopping centers and companies enables decision makers to simulate different spatially explicit development scenarios.

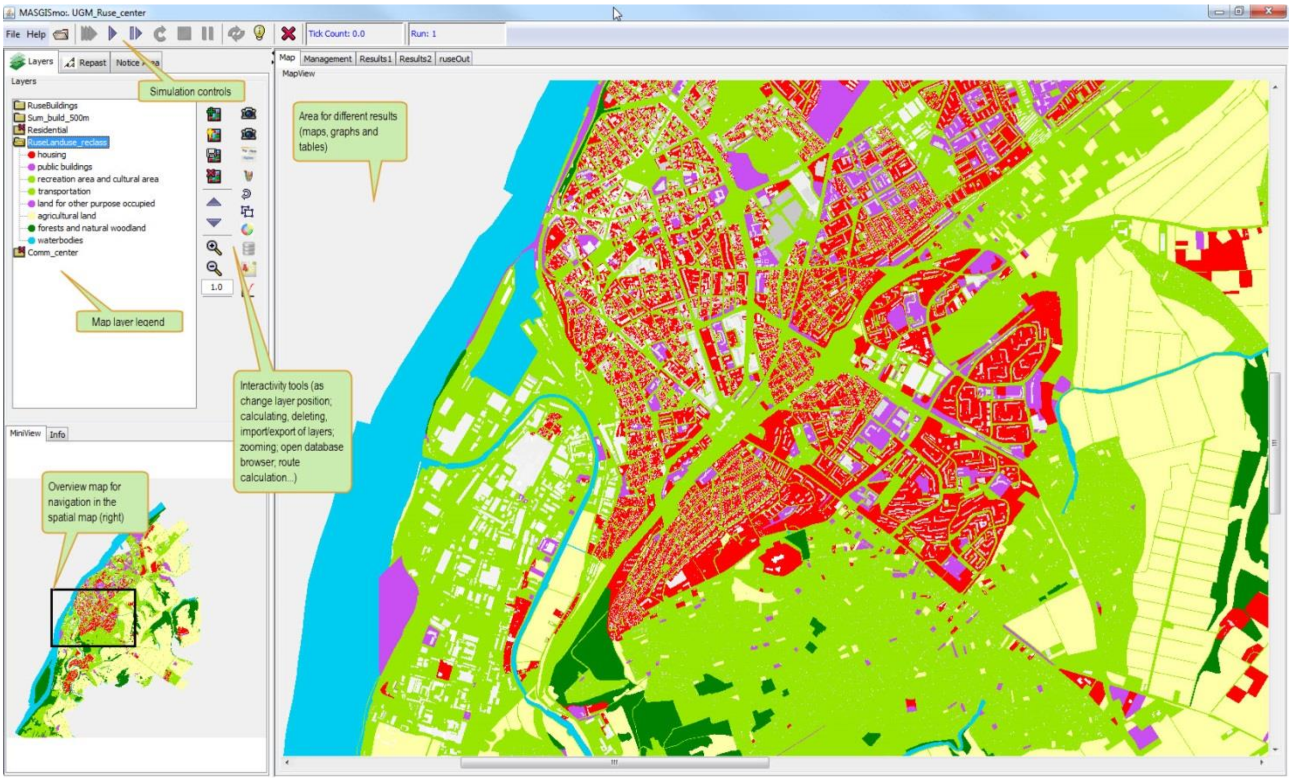


Figure 2.3.1: Screenshot of MASGISmo's UI at an early stage of the urbanAPI project, © AIT

2.3.1.1 MASGISmo basics

The standard graphical user interface (UI) of MASGISmo has been developed during several years. The most important components of the UI are related to different main functionalities which are:

- spatial simulation features conducted through GIS functions,
- interactive control features,
- scenario simulation analysis and
- visualisation features.

2.3.2 MASGISmo's enhancements for the urbanAPI project since 01/2013

The following list summarizes the main changes:

- Integration of GeoServer and OpenLayers to import and export different file formats and generate an interactive web-interface using JavaScript
    - Additional import and export of GIS layers (Vector layers) in different file formats
    - Interaction via Internet for users
- Implementation of an internal web browser
    - Model specific user interaction
    - User tutorial platform
- Basic agent decision rules

2.3.2.1 Integration of GeoServer/OpenLayers and Web-interface

The main client-server architecture has been changed to enable a more advanced user interaction. Compared to the prior architecture, as the following figure shows (see also deliverable D3.2 from 01/13) we have worked on a tighter integration of GeoServer/OpenLayers and PostGIS[2] databases.

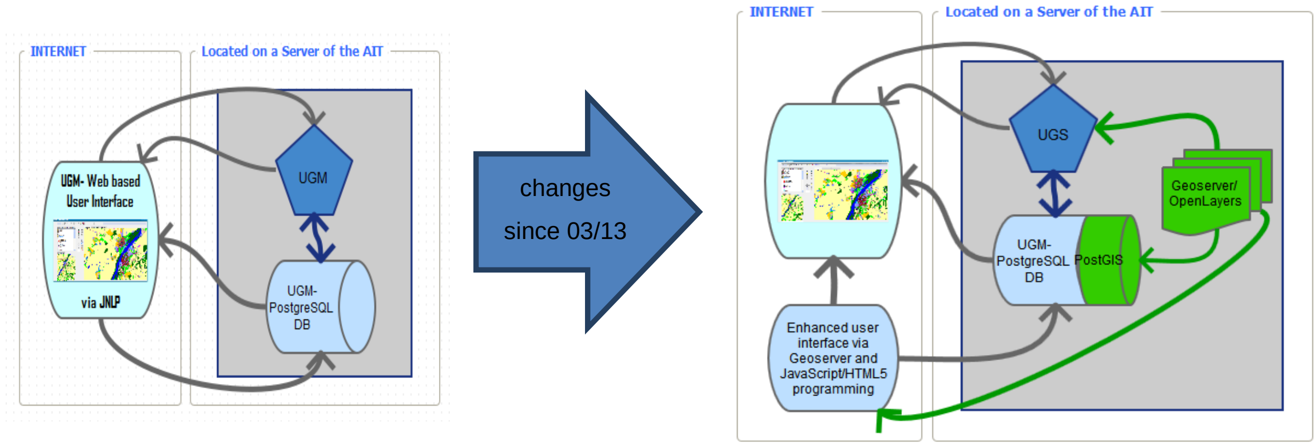


Figure 2.3.2: Changes to the main UGS interaction architecture (left side old -- right side new architecture), © AIT

Java code has been written enabling within the core UGS model to interact with the PostGIS database. This means now it is possible to use GIS vector layers stored in the database and export files in different formats (png,svg, kml, GeoJson) to show them in the UGS core model. Additionally it is possible to create new vector

[2] More information can be found under http://geoserver.org/display/GEOS/Welcome , http://openlayers.org/ and http://postgis.net/ , tested August 2013

layers from the UGS core model and show the results of the simulation in the interactive web interface. The following figure shows an example of this Web interface. The red dots in the figure are simulation results depicting new built up areas. Using OpenLayers and JavaScript-programming enables the user to interact with the simulation by evaluating parameter changes in the core UGS. Scenarios with different agent behaviour will lead to different spatial development for the new built-up areas and the visualization can help to evaluate the scenarios.

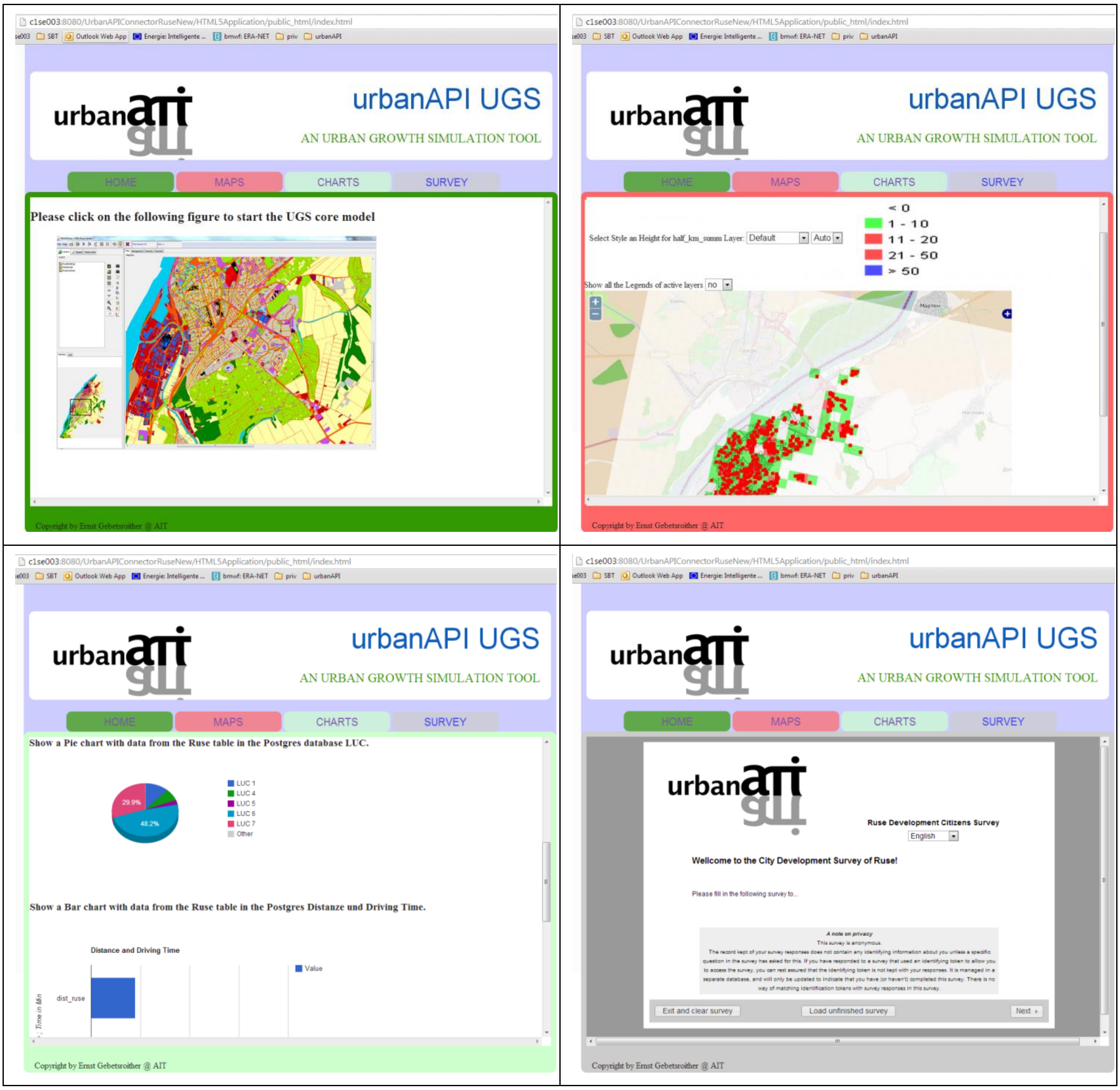


Figure 2.3.3: WEBINTERFACE: **UGS core start page** upper left, **Interactive OpenLayers/GeoServer** upper right, **Interactive PostgreSQL Database Information** lower left, **Ruse online Survey** lower right

All these different services are stored on a local server at AIT (currently not publicly accessible) and will be opened during the first prototype evaluation phase in November.

2.3.2.2 Implementation of an internal web browser

MASGISmo was extended by an internal web browser programmed in Java (see the figure below), enabling the user of the UGS core module (Java program) to directly view information from the web interface described above. Furthermore it enables to interact with local stored information during the simulation session. Thus the figure below refers to the link of the browser addressing local file storage. This storage path can be defined by the user of the UGS core module at the beginning of each simulation run.

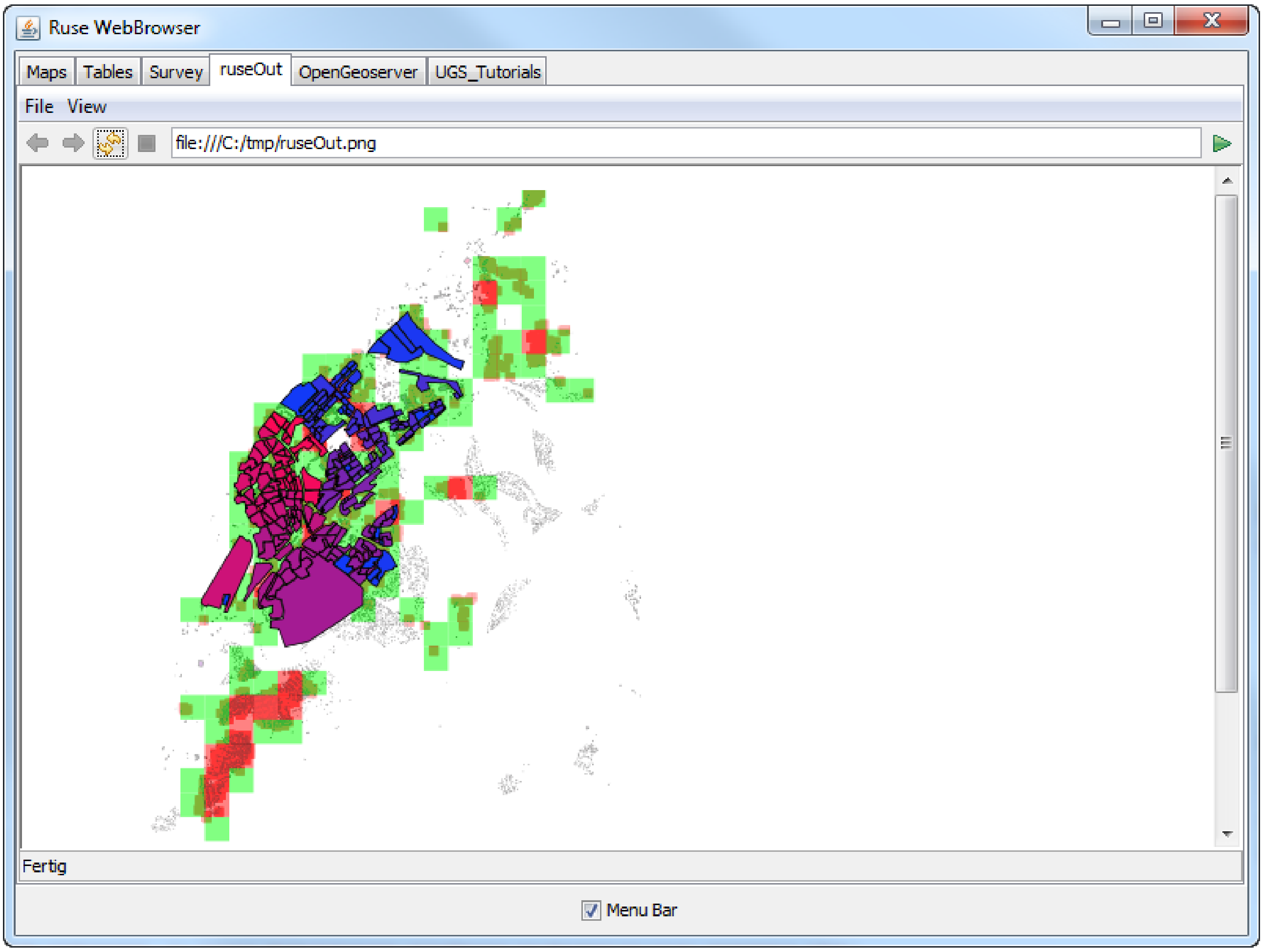


Figure 2.3.4: MASGISmo's internal web browser

The internal web browser is also used to reference to the tutorial material how the UGS can and should be used. The following figure shows part of the tutorial. Beside the main tutorials provided by the UGS developer we would like to implement a discussion board (see figure below right) enabling the user of the

UGS do discuss several problems. The discussion board can also be used to collect the user experience and questions during the evaluation phase of the UGS.

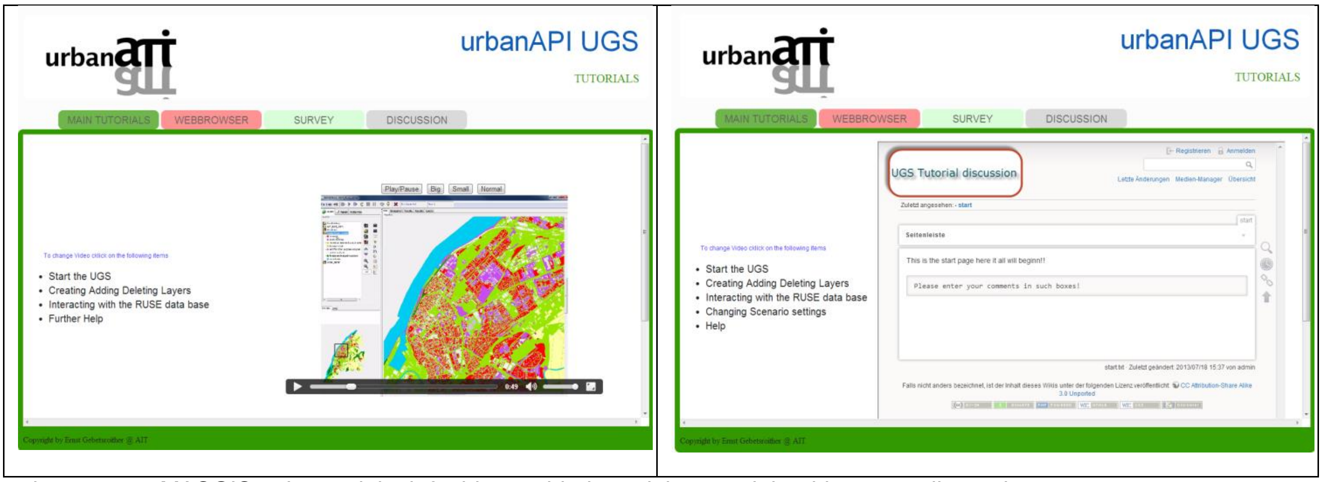
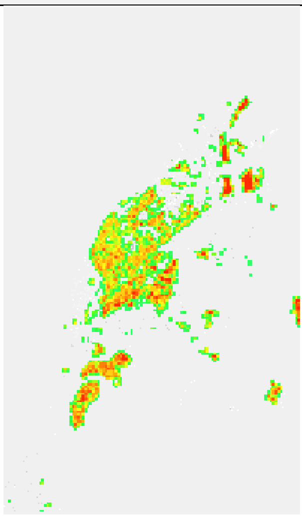
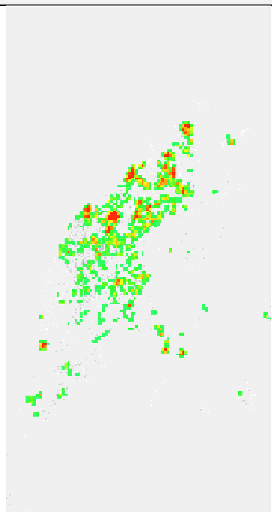
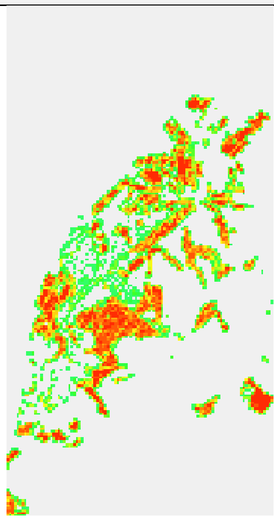
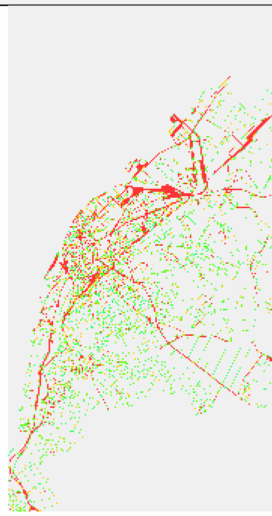
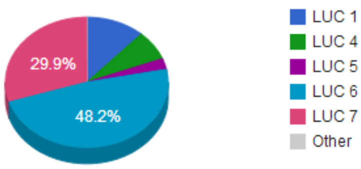
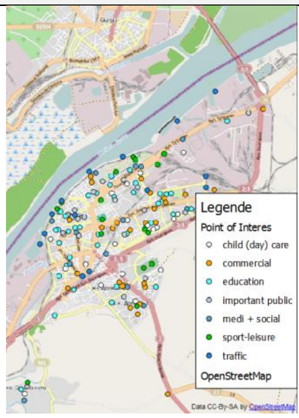


Figure 2.3.5: MASGISmo's tutorials**;** left-side provided tutorial parts, right-side a user discussion WIKI

2.3.2.3 Basic agent decision rules

One of the core tasks for the establishment of an UGS is to analyse available data to find out what the agent decision to move around drives. Often we can use historical data about the movement to analyse the urban development. As we describe in D3.7 in detail the current data situation for Ruse is somehow fuzzy and – as expected - many problems occur when analyzing the content. Thus we are behind our planned schedule for this task. To continue with the development of the basic agent decision rules we used a method combining different preferences of an agent to select a certain area where to move to in the future. On a basis of 100x100m raster cells we calculated several factors which are expected to influence the target area decision. These are e.g. percentage of different land use classes or the driving distance to some Points of Interest (POIs), newly planned commercial areas providing workplaces, sport facilities, recreational areas etc. The following table just shows exemplarily some of the analyzed classes.

Table 2.3.1: Analyses of shares for some land use classes (LUC) on a 100m grid cell (green means low shares red high shares)

LUC housing	LUC 2 public buildings	LUC 3 recreating and cultural areas	LUC 4 transportation
			
LUC shares for a certain cell (available for each cell)	POIs		
			

2.3.2.4 Land use dynamics simulator

The dynamics of the agents within the UGS is mainly driven by their individual preferences and boundary conditions. Boundary conditions are e.g. available space for buildings, rules for the allowed building types (height etc.) in certain areas within the city. These conditions are mainly determined by the local authorities within their master plans and urban development concepts. They can be also part of scenarios and can be changed in the UGS. The preferences for the agents are difficult to estimate. Currently no information about them exists, but there are mainly three ways how we can tackle this problem.

a) Analyzing historical agent behaviour with e.g.:
    - correlation analyses of land use patterns
    - correlation analyses of actor movements
b) Expert knowledge from e.g.:
    - local experts or planning authorities assume
    - If - then analyses
c) Collecting information about user preferences by e.g.:
    - quantitative/ semi-qualitative Interviews
    - focus group workshops or
    - online surveys

For the UGS we can and will use some of them. As far as possible we try to get information from existing historical data which is very difficult, because of data gaps (see D3.7.). We use the expert knowledge of our project partners in Ruse and ASDE. Last but not least the intended online survey will gather information which is fed in the UGS simulation in two ways:

- The first is that the citizens can vote for certain area in the city, which is result of all their preferences and background conditions.
- The second is that the online survey asks for their preferences und this leads to an average preference distribution. This distribution can be classified by certain socio-economic parameters as e.g. age or education, unemployment or living standard.

The UGS uses this information in the Agent behaviour settings see figure 2.3.6 below. Currently only eight preferences are assumed to be important on the one hand and can be quantified with the existing data on the other hand. This number of Agent preference parameters can be changed by a later stage of the model development and furthermore some parameters can be made inactive by setting them to zero until the spatial information is available (e.g. rental prices)

- Agent behaviour parameters

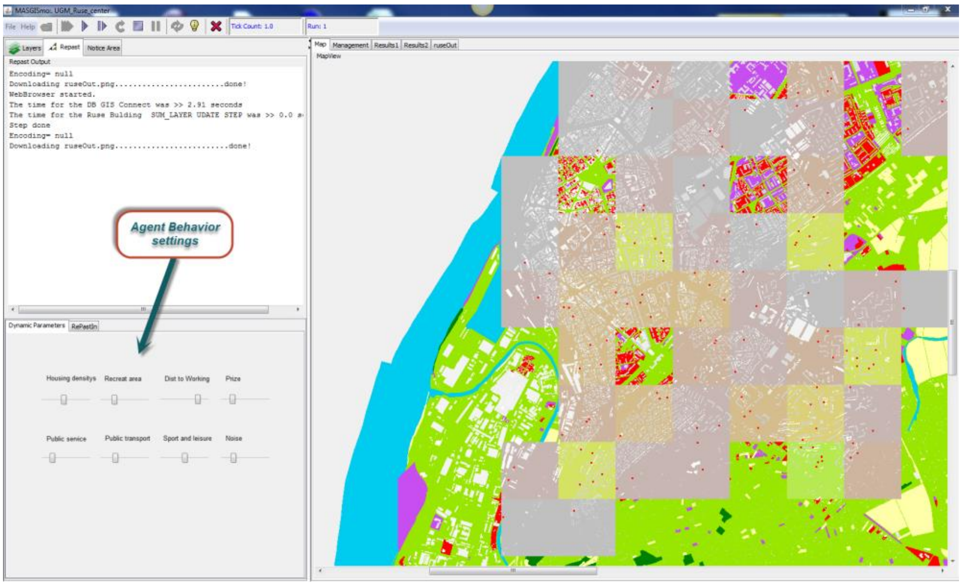


Figure 2.3.6: Interface for dynamic Agent behaviour settings via input slider

2.3.2.5 Land use dynamics / impact explorer

Similar to the above presented maps (Table 2.3.1) other important parameters will be calculated to show the impact of the urban development. The available data e.g. enables roughly to estimate the energy demand based on building age and an assumed average energy consumption per age class on a 500x500m grid cell level. Furthermore it is planned to approximate the changes of traffic emissions due to the Urban Growth development, but until know it is not clear if we will get enough reliable data for this task.

- General Map algebra functions:
  - aggregation to different grid cell resolutions,
  - block sum, block average
  - focal share
  - distances to certain Points of Interest

The UGS graphical user interface developed with MASGISmo enables the user to derive specific analyses[3].

[3] Tutorial videos will show how this works.

2.3.3 Graphical user interface

The UGS has different components interacting (see figure 2.3.2 right). All the different parts can be managed by the UGS core module (the Java program). Import and export function to and from the different components enable to retrieve a lot of different data and data formats. Via the GeoServer exporter (a new developed part of the UGS core module) ESRI shapefiles, image files or KML files and others can be extracted. These files can be used within other tools as e.g. QGIS[4]. One feature of the Database Browser, another main part of the UGS core module, enables to interactively create new charts from data stored in the database, see figure 2.3.6 below.

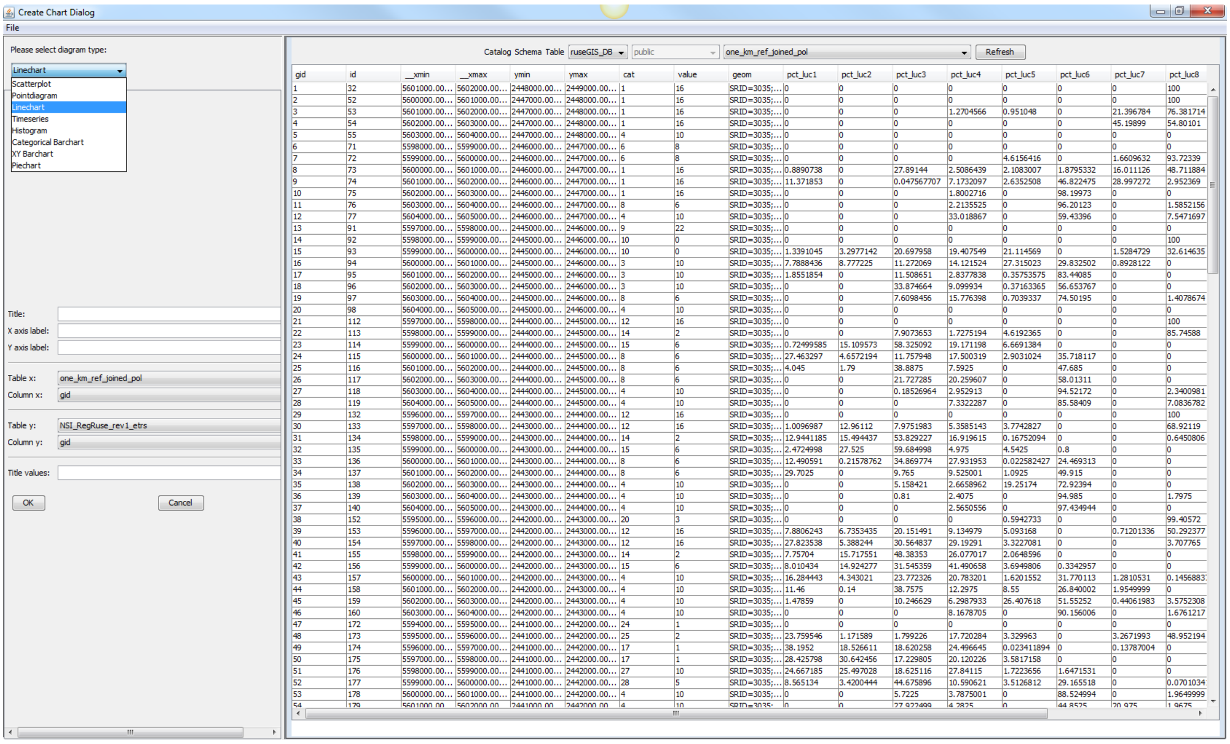


Figure 2.3.6: *Chart Generator Dialog* of the MASGISmo-DatabaseBrowser

Together with IGD-Fraunhofer a 3D-visualization is under development applying X3DOM, which can be viewed within a normal web browser. An example depicts Table 2.3.1 left. This will be part of the urbanAPI internet portal described in D3.6. As standalone application we are additionally trying to visualize simulation results with Google Earth, but currently this faces performance problems, at least on a normal laptop when trying to show all the buildings in Ruse in 3D.

---

[4] www.qgis.org

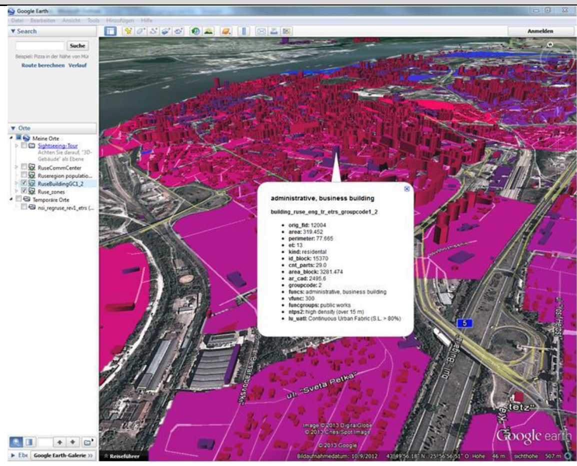
3D-Visualization with X3DOM	Google Earth 3D-Visualisation
	

Figure 2.3.7: *3D-Visualisation results so far*

References

Akkermans, H.: Emergent Supply Networks: System Dynamics Simulation of Adaptive Supply Agents. In: System Sciences, Proceedings of the 34th Annual Hawaii International Conference on System Sciences vol., no., pp.11 (2001)

MASGIGmo, http://systemsresearch.ac.at/exchange/gebetsroither/Tutorial_von_MASGISmo/Welcome.html

Pourdehnad, J., Maani, K., Sedehi, H.: System Dynamics and Intelligent Agent-Based Simulation: Where is the Synergy. In: Proceedings of the 20th International Conference of the System Dynamics Society (2002)

Schieritz, N., Groessler, A.: Emergent Structures in Supply Chains: A Study Integrating Agent-Based and System Dynamics Modeling. In: Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS'03) (2003)

Schieritz, N., Milling, P.: Modeling the Forest or Modeling the Trees. In: Proceedings of the 21st International Conference of the System Dynamics Society (2003)

Scholl, H.J.: Looking Across the Fence: Comparing findings from SD Modeling Efforts with those of Other Modeling Techniques. In: Proceedings of the 19th International Conference of the System Dynamics Society, Atlanta, GA (2001a)

Scholl, H.J.: Agent-Based and System Dynamics Modeling: A Call for Cross Study and Joint Research. In: Proceedings of the 34th Annual Hawaii International Conference on System Sciences, pp. 1-8 (2001b)

Tomlin, Dana C. 1983. Digital cartographic modeling techniques in environmental planning. Yale University, School of Forestry and Environmental Planning, New Haven, Connecticut, 298 pp.