EEPOS - Energy management and decision support systems for energy positive neighbourhoods

Co-funded by the European Commission within the 7th Framework Programme.
Grant Agreement no: 600050.
2012-10-01…2015-09-30 (36 months).

Results of German/Austrian demonstration

Authors:

Florian Judex .... AIT
Gerhard Zucker............................. AIT
Filip Petrushevski .... AIT
Stefan Hauer .... AIT
Mario Köstl .... AIT
Max Blöchle .... AIT

Table of contents

1. EXECUTIVE SUMMARY

The aim of this document is to present the EEPOS German/Austrian demonstration. Based on the specifications of the dwellings in Langenfeld, a virtual demonstrator was created that allows to experiment with the available demand response potentials. The virtual demonstrator uses real world dwellings as a foundation which allows their operational data to be analysed for the potential to optimize neighbourhood energy management using the EEPOS IT Platform. This demonstrator uses dynamic thermal simulation tools that are used for providing qualitative information on the thermal performance of energy systems and buildings. The goal of the virtual demonstrator is to prove the methodology and show replicability of the solution.

The document describes the creation of models and acquisition of data required for the implementation of the demonstrator. It then explains the utilized control algorithm and the communication and operating procedure between the individual components into a working demonstrator. It concludes with experiments and results that demonstrate the working of the demonstrator.

The document concludes that the combination of EEPOS platform together with the simulation environment (Chapter 4.1) shows effective operation (Chapter 5.3) and can be the baseline for further investigations. The control algorithm generated equally distributed values to effectively shift loads towards the daily mean (thus reducing peak loads), as analysis (Chapter 5.4) showed.

The demonstrator successfully displayed the mode of operation and positive effect of neighbourhood energy management in a simulated environment. Detailed CO2 savings and peak load reductions can be seen D4.5.

2. INTRODUCTION

2.1 Purpose and Target Group

The purpose of this document is to describe how the technical solutions, developed in WP2 and WP3, are combined into the EEPOS demonstrator. The used technologies and tools are named and the implementation of the demonstrator is described. This report is used for documentation and validation purposes of the demonstrator's detailed workings. Furthermore, this deliverable will be a helpful source for other public deliverables as it provides an in depth view of the demonstrator.

2.2 Contributions of Partners

Due to the change of the partner from ennovatis to DERlab during the demonstration execution, DERlab provided support during the extension of EEPOS platform that is detailed in D4.5 in more detail. AIT took over the task lead and was responsible for the data and model provision as well as the coding of the glue logic that combines the EEPOS platform to the simulation environment. As such AIT was also responsible to demonstrate the working of the demonstrator. The work was carried out in close cooperation with DERlab and other partners providing valuable feedback.

2.3 Relations to Other Activities

This document strongly builds on the outcomes of prior work packages, as the demonstrator could not have been realized without building on top of them. Furthermore, there is a strong connection to D4.5 that shows the detailed CO2 savings and peak load reductions.

3. Data Acquisition and Modelling

This chapter describes the creation of models and generation of data required for the implementation of the demonstrator in Chapter 4.

3.1 Data Acquisition

3.1.1 Building Energy Data

To be able to model the German demonstrator in adequate quality each building model has to be parameterized individually in terms of its physical properties resulting in a realistic thermal behaviour of the building. This characteristic behaviour is effected by the energy loses of the building through the building envelope specified by material data and its internal gains specified by the usage of the building. Based on the year of construction, the building usage and the location, national norms and standards can be used for parametrizing the building.

Based on satellite images from the demonstrator region the overall construction year of the buildings is estimated between 1958 and 1968. Their main category is single- or multifamily house. Based on this estimation the parameters as seen in Table 1 below can be specified for the building models.

Table 1 Overview of physical building parameters

Parameter	Value	Unit	Information Source
u-value of the outer walls	1.4	W/(m² K)	EnEV 2009 / “Tabelle 2-3”
u-value of the roof	1.4	W/(m² K)	EnEV 2009 / “Tabelle 2-3”
u-value of the ground floor	1.0	W/(m² K)	EnEV 2009 / “Tabelle 2-3”
u-value of the window	2.7	W/(m² K)	EnEV 2009 / “Tabelle 2”
g-value of the window	0.75		EnEV 2009 / “Tabelle 2”
Window - floor ratio	20	%	EnEV 2009
Overall building construction	medium		Assumption
Heating set point	20	°C	DIN V 18599-10 / “Tabelle 3” DIN V 18599-10 / Annex A
Internal gain light	7	W/m²	DIN V 18599-10 / Annex A
Internal gain plug load	4	W/m²	DIN V 18599-10 / Annex A
Internal gain people	70	W/person	DIN V 18599-10 / Annex A

Being able to model the usage of the buildings the internal gains represented as schedules need to be defined for the category single or multifamily house. Due to missing information about the exact occupancy of each building an assumption had to be made. By combining “DIN V 18599-10 / Tabelle 3” and “ÖNORM B 8110-5:2011, Tabelle 2” it was possible to define an hourly profile. Taking both parameters (building physics data and dynamic behaviour of occupancy) into account, it is possible to define the thermal behaviour of the demonstrator test buildings.

3.1.2 Building Geometry Data

The goal of this chapter is to describe the creation of geometrical building data as an input for modelling the thermal behaviour of different buildings within an urban quarter. Moving from a single building approach to a city-wide one (or at least an urban quarter or district) but still having the buildings as smallest reference unit, is nowadays still a big challenge. First, because a city is a more complex system than a single building: As it is not just a “simple” aggregation of buildings, many more entities and their mutual relations must be considered. Secondly – and more profoundly – because obtaining such city-wide information is often difficult due to inaccurate or just not integrated data. The most common way to create such data sets is through the use of a *Geographic Information System (GIS).* A GIS is a system designed to capture, store, manipulate, analyse, manage and display all types of geographical data.

GIS is a relatively broad term that can refer to a number of different technologies, processes and methods. It is attached to many operations and has many applications related to planning, engineering, management, transport/logistics, insurance, telecommunications, and business. Generally, a GIS is custom-designed for an organization. Hence, a GIS developed for a special application or purpose may not be necessarily interoperable or compatible with a GIS that has been developed for some other applications. Today there are several worldwide used business platforms and some free, open-source GIS packages, which run on a range of operating systems and can be customized to perform specific tasks. Very often used GIS are Intergraph, Autodesk (best known for its AutoCAD family) and the open-source software GRASS (Geographic Resources Analysis Support System) used for geospatial data management and analysis, image processing, spatial modelling and map production. Other important GIS are MapWindow GIS, SAGA GIS and uDig.

In the recent years *QGIS* got increasingly popular (previously known as "Quantum GIS" [1]. It is a cross-platform free and open-source desktop GIS application that provides data viewing, editing, and analysis capabilities. QGIS also provides the integration with other open-source GIS packages, including PostGIS, GRASS, and MapServer to give users extensive functionality. So called plugins, written in Python or C++, extend the capabilities. Plugins exist to geocode using the Google Geocoding API, to perform geoprocessing (fTools) similar to the standard tools found in ArcGIS, and to interface with PostgreSQL/PostGIS, SpatiaLite and MySQL databases. QGIS runs on multiple operating systems including Mac OS X, Linux, UNIX, and Microsoft Windows and allows the use of most common data formats such as *dxf*, *shapefile*, *coverage*, and *personal geodatabase*. Additionally *MapInfo*, *PostGIS*, and a number of other formats are supported in QGIS. As an available for free software QGIS can be modified by everyone to perform different or more specialized tasks. All this leads to the rapidly increasing importance of QGIS. Today almost a number of public and private organizations have adopted QGIS. Among others, these are the Austrian state Vorarlberg, and the Swiss Canton of Solothurn.

Although QGIS is very popular (partly due to the fact of being free of cost) the most prominent GIS is *ESRI ArcGIS*). ArcGIS is proprietary commercial software and pretty expensive, nevertheless it is the most common used GIS in academic and commercial geo-informatics. For most applications it is still state-of-the-art. Many GIS innovations like on-the-fly map projecting as well as data formats (e.g. *shapefile* and the *personal geodatabase* format) where originally developed by ESRI. ArcGIS (version 10.0 and higher) consists of GIS products running on desktop, as well as GIS products that run on a server or on mobile devices. Its G*raphical User Interface (GUI)* application *ArcMap* enables mapping, analysing and visualization of almost all existing data formats. Figure 1 shows the ArcGIS GUI, its most relevant components and applications like *ArcToolbox* (which includes several GIS tools grouped to functional classes) and *ArcCatalog* (used to access, browse, organize, and manage data files).

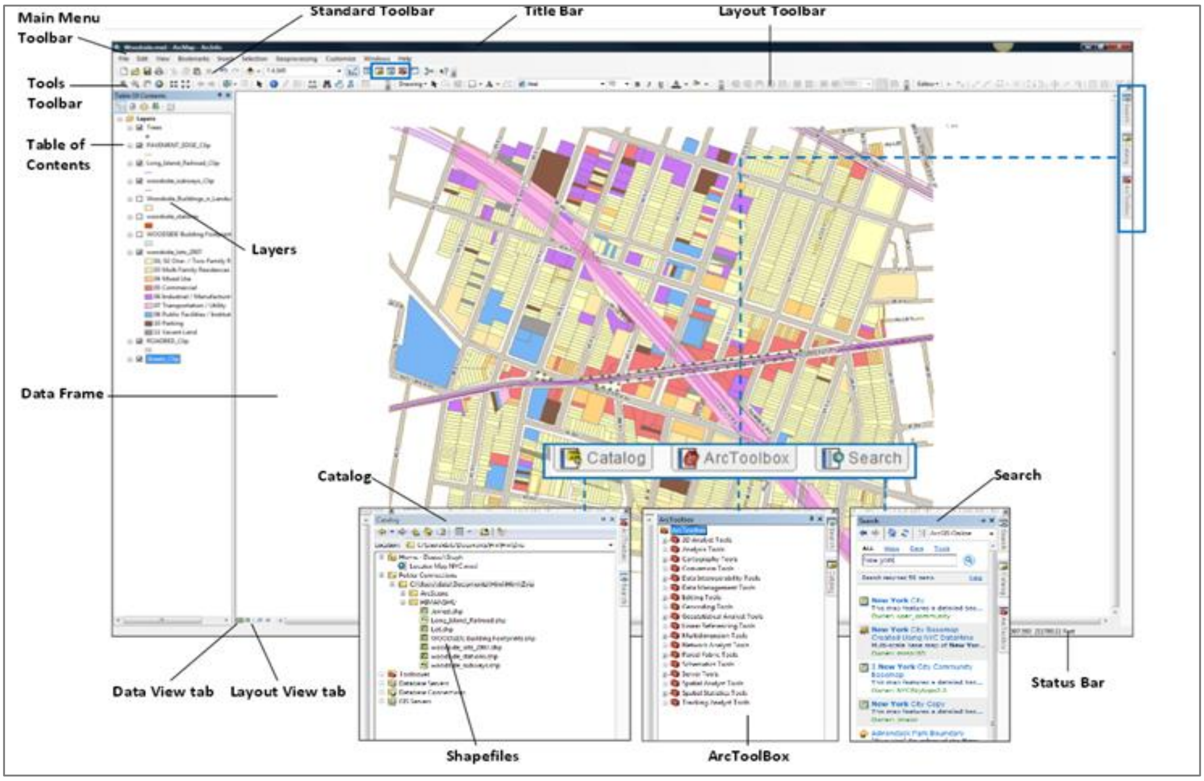


Figure 1. ArcMap as an example of a common GIS GUI

ArcGIS Desktop includes customization options and third-party extensions that can be added to ArcGIS providing added functionality, including 3D Analyst, Spatial Analyst, Network Analyst, Survey Analyst, Tracking Analyst and Geostatistical Analyst. Similar to other software GIS it allows users to create maps and data sets with many layers using different map projections. Maps can be assembled in different formats and for different uses. Different kinds of raster images are supported and the software can perform geo referencing of images. It also uses a spatial reference to project features on the fly and uses the linear units defined by the map projection to measure distances and areas. We at AIT use for our research projects ArcGIS 10.x as well as QGIS.

Generally all GIS model geographic information as a logical set of layers or themes. These themes can contain data layers for administrative areas, land-use areas representing vegetation or parcel representing landownership. Besides "*polygon*" information it can also incorporate point information (e.g. addresses), lines (e.g. streets, rivers, pipelines and even complex networks with specified rules) and so called raster datasets. Raster files are cell-based datasets used to hold imagery, digital elevation models or other thematic data. Even complex attributes and descriptive information traditionally stored as tabular information is used to describe features and categories about the geographic objects within each dataset.

A GIS not only uses maps to visualize features and feature relationships to work with the geographic information. It normally also includes a large set of *geo-processing* functions to take information from existing datasets, apply analytic functions and write results into new result datasets. GIS applications are tools that allow users to create interactive queries (user-created searches), analyse spatial information, edit data in maps, and present the results of all these operations. Modern GIS technologies use digital information, for which various digitized data creation methods are used.

The common method of data creation is digitization, where a hard copy map or survey plan is transferred into a digital medium. This can be done in most cases directly via special editing tools within the GIS platforms, but in many cases it is done through the use of *Computer-aided design (CAD)* programs and geo-referencing capabilities. On the other hand many enterprises, governments or urban administrations provide digital information in various formats, and even several open GIS communities create editable data for free use. In most cases – especially when handling with a huge number of objects – the use of such data sets will decrease the amount of creating digital data enormously.

A nowadays very often used data set is *OpenStreetMap®*. *OpenStreetMap (OSM)* is a collaborative project to create a free editable map of the world. One is free to copy, distribute, transmit and adapt these data, as long as one credits it with "©OpenStreetMap.org contributors". As the OSM project is aimed at creating a free, world-wide geographic data set, it aims to be for geo data what Wikipedia is for encyclopaedic knowledge, so everyone can participate (including all problems like quality and truth known from Wikipedia). The focus is mainly on transport infrastructure (streets, paths, railways, rivers), but OSM also collects a multitude of points of interest, buildings, natural features and land use information, as well as coastlines and administrative boundaries.

OSM relies mostly on data collected by project members using their GPS devices and entered into the central database with specialized editors. For some areas, third party data has been imported. OSM data quality and coverage differ between regions. Many European cities are covered to a level of detail that surpasses what proprietary data vendors have to offer. Often, OSM will also be the first to have a new housing development or a new motorway exit mapped. But in some, mostly rural areas there might be nothing in the database except some primary roads. More about the OSM project and how one can use its data can be found at the OSM Wiki homepage [3].

There are several ways to extract and download a specified area. An easy and convenient procedure to do this is using *BBBike extracts* [4]. This site allows one to extract areas from *Planet.osm* database in *OSM*, *CSV*, *ESRI shapefile* or several other common formats. Planet.osm is the OpenStreetMap data in one file: all the nodes, ways and relations that make up our map. A new version is released every week. It is a big file (XML variant over 576.6GB uncompressed or 42GB bz2 compressed at 2015/05/19). The maximum area size using *BBBike extracts* is 24,000,000 square km, or 768MB file size. The user just has to move the map to the desired location, create a boundary box and select a desired file format. At the end one has to enter an email address and a name for the area to extract. After a few minutes one gets a notification by e-mail where to find the extract ready for download.

As we wanted to edit the downloaded building polygons before using them in the dynamic simulation environment *EnergyPlus*, we opted for the *ESRI shapefile* format, which is a common standard for representing geospatial vector data. OSM uses a topological data structure for their vector data, with four core elements (also known as data primitives):

- ***Nodes*** are points with a geographic position, stored as coordinates (pairs of latitude and longitude) according to WGS 84. Outside of their usage in ways, they are used to represent map features without a size, such as points of interest or mountain peaks.
- ***Ways*** are ordered lists of nodes, representing a polyline, or possibly a polygon if they form a closed loop. They are used both for representing linear features such as streets and rivers, and areas, like forests, parks, parking areas and lakes.
- ***Relations*** are ordered lists of nodes, ways and relations (together called "members"), where each member can optionally have a "role" (a string). Relations are used for representing the relationship of existing nodes and ways. Examples include turn

restrictions on roads, routes that span several existing ways (for instance, a long-distance motorway), and areas with holes.

- ***Tags*** are key-value pairs (both arbitrary strings). They are used to store metadata about the map objects (such as their type, name and physical properties). Tags are not free-standing, but are always attached to an object: to a node, a way or a relation.

The shapefile is a simple, non-topological format for storing the geometry and attribute information of geographic features. It defines the location and attributes of geographically referenced features in three (or more) files with specific file extensions. Three of them are at least required to work with a shapefile: The .shp is the main file that stores the feature geometry; .dbf a dBASE table that stores the attribute information and .shx is the index file that stores the index of the feature geometry to connect the .shp and .dbf file.

Figure 2 displays the *BBBike extracts* download specifications for the test area Martinstraße in Langenfeld (Rheinland).

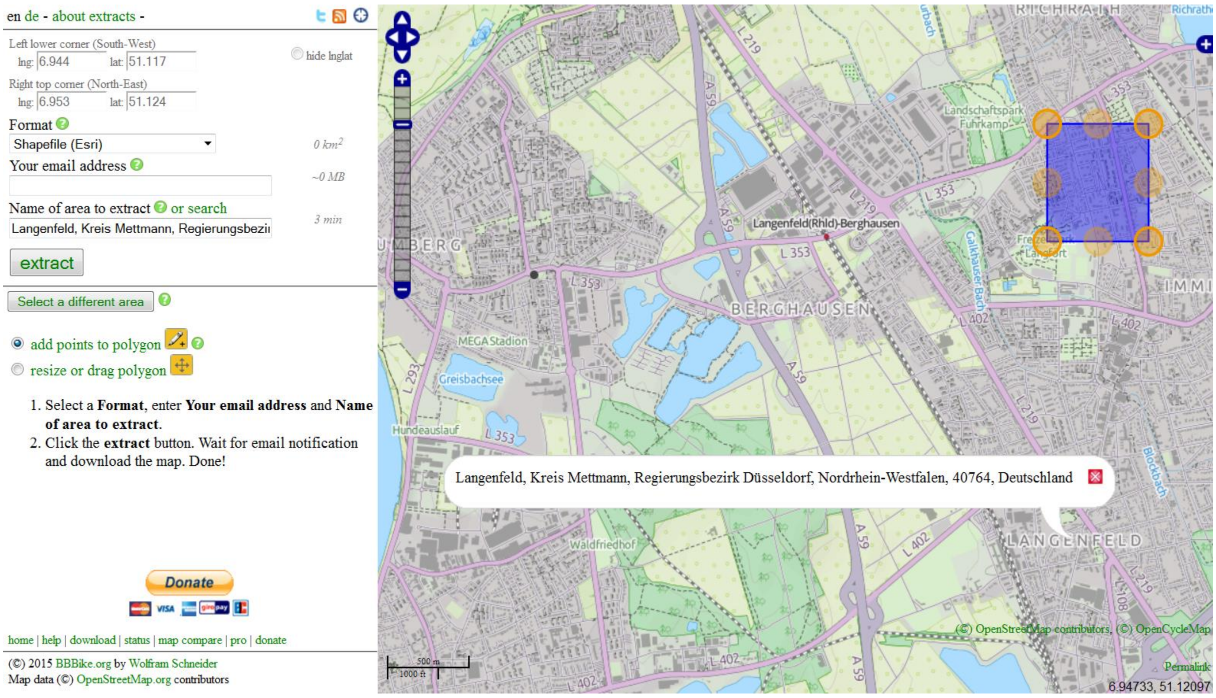


Figure 2. BBBike extracts download specifications for the preferred test area

Depending on the area and data collected by project members the OSM data can differ in amount, quality and level of detail. In some areas only street data exist (but these in a very good quality!); in other parts of the world the whole data set is available at the highest level of detail. For Langenfeld the data set includes buildings (in which we were primarily interested) and streets. After downloading one can import the data into the GIS and start analysing and editing the building objects.

All OSM vector data are projected in the geographical *Coordinate Reference System (CRS)* WGS 84. The *World Geodetic System (WGS)* is a standard for use in cartography, geodesy and satellite navigation. It comprises a standard coordinate system for the Earth. The actual revision is WGS 84 (EPSG 4326), which is also the CRS used by the *Global Positioning System (GPS)*. Data stored in this projection use geographical longitude and latitude in decimal degrees. So all calculated lengths and areas will also be in decimal degree units. In order to calculate building metrics like area and perimeter in suitable units like [m] and [m²] one has to re-project the downloaded polygons. The chosen CRS for this was ETRS89-LAEA Europe, also known in the EPSG Geodetic Parameter Dataset under EPSG 3035. The *European Terrestrial Reference System 1989 (ETRS89)* is an ECEF (Earth-Centred, Earth-

Fixed) geodetic Cartesian reference frame for all Europe. The Lambert Azimuthal Equal Area (LAEA) projection is centred at 10°E, 52°N. Coordinates are based on a false Easting of 4321000 meters and a false Northing of 3210000 meters. Being based on an equal area projection, this CRS is suitable for generalizing data, statistical mapping and analytical work whenever a true area representation is required. ETRS89-LAEA Europe is the official CRS used by all EU agencies. For example the *European Environment Agency (EEA)* reference grids and Urban Atlas Data, as well as *Eurostat Nomenclature of Units for Territorial Statistics (NUTS)* regions are provided in this projection.

Figure 3 shows the location of the re-projected buildings within the municipality of Langenfeld. Additionally to the OSM data the municipality borders of Langenfeld and an available basemap as background were added.

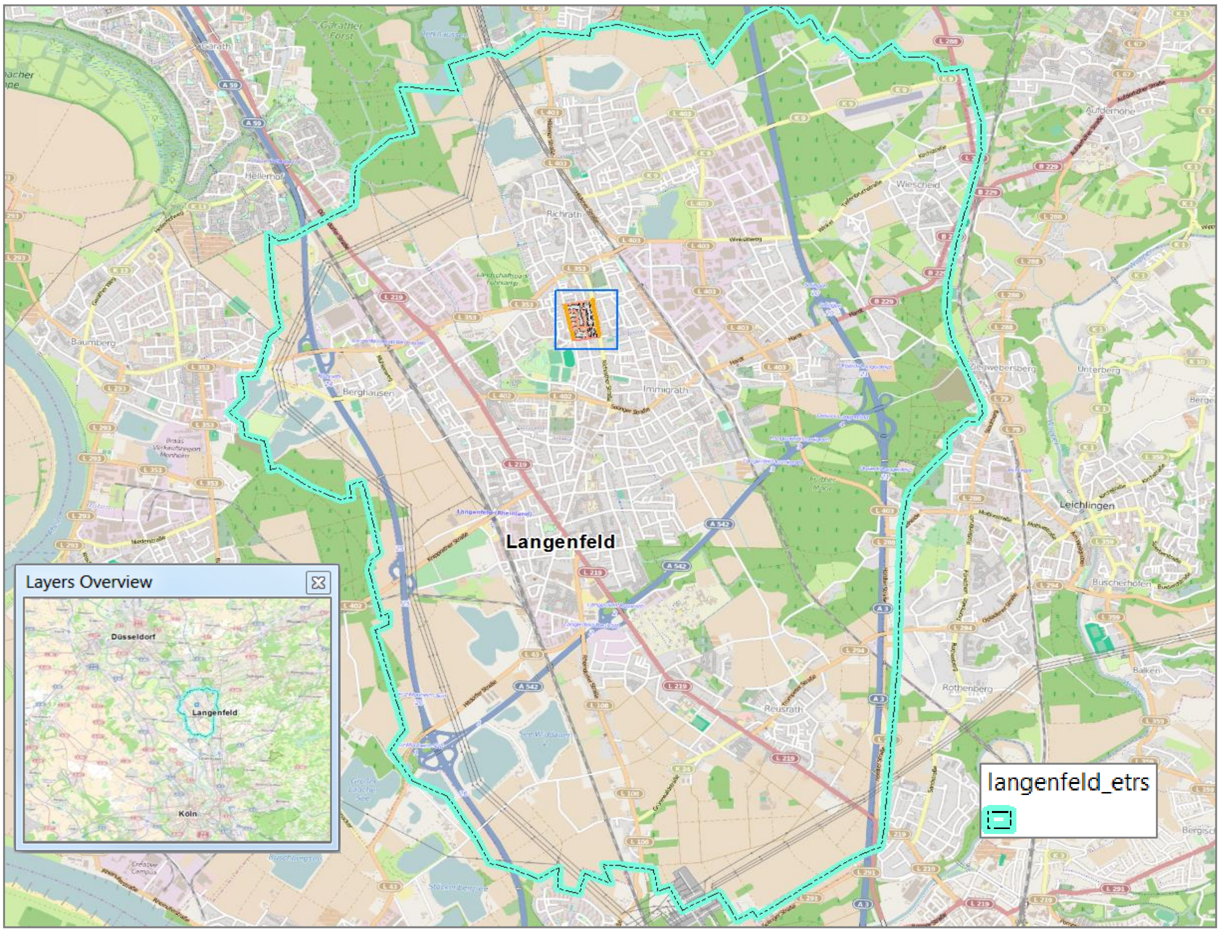


Figure 3. Location of the test area buildings within the city of Langenfeld using ArcGIS 10.0

As OSM data are collected by project members using their GPS devices and entered into the central database with specialized editors, it is recommended to check the data for their meaningfulness and it is advisable to check on meaningfulness and completeness. Sometimes it is necessary to digitize some missing buildings or to delete not anymore existing objects. In some special cases one has to cut an object into two or more parts, as the OSM object contains different uses – e.g. a garage and a residential part. To do so one has to start a so called *edit session*. During an edit session, you can create or modify vector features or tabular attribute information. When you want to edit, you need to start an edit session, which you end when you're done. Edits are temporary until you choose to save and apply them permanently to your data.

Figure 4 illustrates some necessary editing processes near the crossing between Jahnstraße and Paulstraße (in the lower left corner of the test area) to improve the OSM input: The original OSM object in the upper left part is too large and has to be cut into two parts – a residential building and a garage. The two objects in the lower part don't exist anymore and had to be deleted. Instead of the left one a new building was built, so it had to be digitized manually.

Figure 4. Examples of some necessary editing processes within the preferred test area

Calculating area and perimeter of polygons within ArcGIS 10.x is an easy thing to do using the Calculate Geometry dialog box. Just create new attribute fields named Area and Perimeter within the attribute table of the feature layer, then right-click the field heading for which you want to make a calculation and click Calculate Geometry. For the calculation of other metric parameters like length, wide or main angle (ie the angle of the main axis of a building in degrees from the geographic North) one needs special tool or extentions – most of them free, some are chargeable. Out of these parameters one can calculate other metric parameters like compactness (an indicator of polygon shape complexity) or aspect ratio (the proportional relationship between width and length of a building).

Additionally, one can create an address-attribute containing street name and house number. Normally one needs for this a separate point layer as input containing this information. With a SPATIAL JOIN in a GIS one can easily connect this information with the buildings. But even if this information is available, it is in many cases very time consuming as many points have inaccurate or wrong XY-coordinates. For Langenfeld there was no such information, nevertheless for a rather small quarter like in our showcase it is normally no problem to generate the addresses within few minutes to hours. There are some useful GIS plugins like GeoSearch in QGIS with which one can easily get the address information. Interesting is also the Nominatim tool (from the Latin, 'by name') to search OSM data by name or address and to generate synthetic addresses of OSM points (reverse geocoding). It can be found at [5]. Of course, for a relatively small quarter there is also the possibility to use an adequate basemap including the addresses and add these addresses manually to the building polygons.

The Figure 5 below shows the result of all above mentioned calculations and steps. Additionally, two rings of interest around the examined quarter have been created.

Figure 5. 2D representation of the urban quarter Martinstraße in ArcMAP

As shown above, a 2D representation for a quarter out of OSM data is no big challenge. The most crucial thing is getting any 3D information as OSM doesn’t provide this important information. On the other hand height and number of levels are significant for modelling the behavior of a building. As long as such information is not included in the buildings like in official polygon data provided by many city authorities - nowadays more and more based on *CityGML* [6] - or present by detailed *Computer-aided drafting (CAD*) input for some selected buildings, one has to obtain the needed information by other sources.

A common way to get building 3D information is by the use of additional digital information out of laser scanning data from the surface. With standard GIS tools one can combine these data with existing building footprints to generate height related information. But for Langenfeld (as for most areas of interest in Europe) such data are not available. Nevertheless there are some other ways to get information about height and number of levels.

An easy way is using *Google Maps* [7] or its download application Google Earth. Both provide for many urban regions complete 3D representations including textured 3D building models which enable one to customize the needed information. Of course, the larger a study area - and therefor the number of buildings with different heights - the more time consuming is such (manual) extraction. Nevertheless it’s better than nothing. Unfortunately there is no such 3D representation for Langenfeld, but both applications offer a huge number of georeferenced photos which ca be very helpful. The following Figure 6 and Figure 7 show two examples of such photos within the quarter Martinstraße uploaded by some Google users.

Figure 6. Richrather Str. 108-112, looking Northwest (source: https://www.google.de/maps/)

Figure 7. Jahnstr. 51, looking East (source: Google Earth via http://www.panoramio.com/)

Another possibility to estimate the desired heights of buildings is Bing Maps 3D (http://www.bing.com/maps/), shown in the next Figure 8. This tool provides 3D representations of most European and American cities. The Bird's-eye view displays aerial imagery captured from low-flying aircraft. Unlike the top-down aerial view captured by satellite, Bird's-eye images are taken at an oblique 45-degree angle, showing the sides and roofs of buildings giving better depth perception for geography.

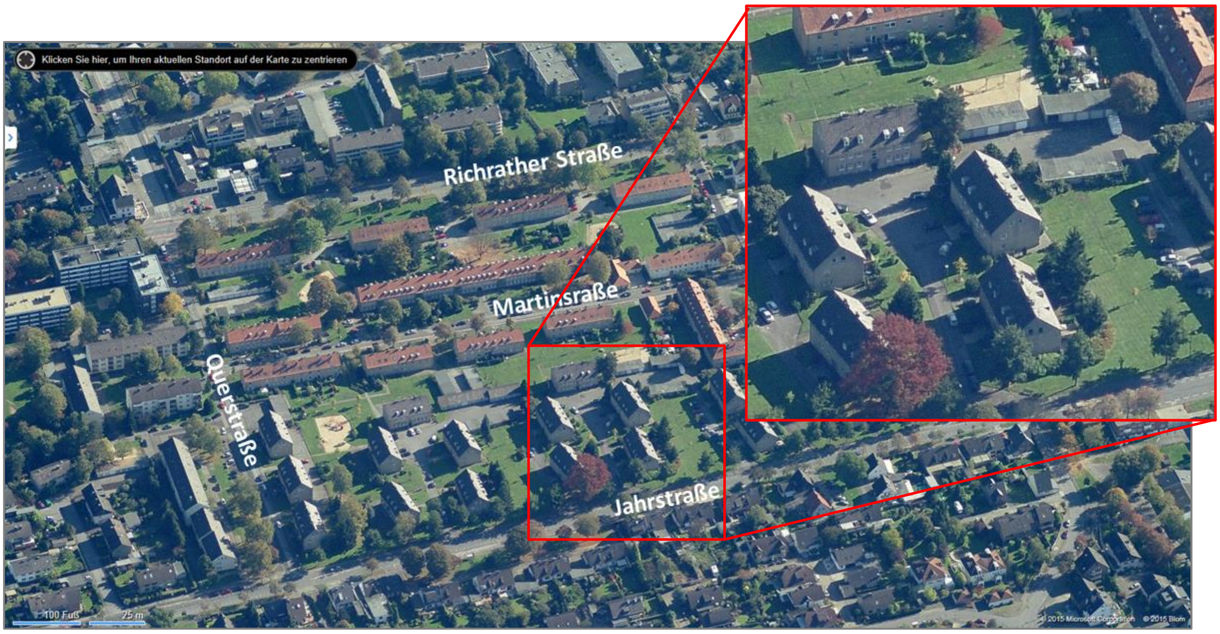


Figure 8. Quarter Martinstraße looking East (source: http://www.bing.com/maps)

Again, for a larger or very heterogenic quarter, district (or even whole city) a manual extraction of building height is probably too time consuming, but as in our show case most buildings are quite similar (even this is a meaningful determination!), so the effort required was reasonable. One just selects all the buildings with same properties and calculates the (added) attributes Height and Level by simple standard GIS operations.

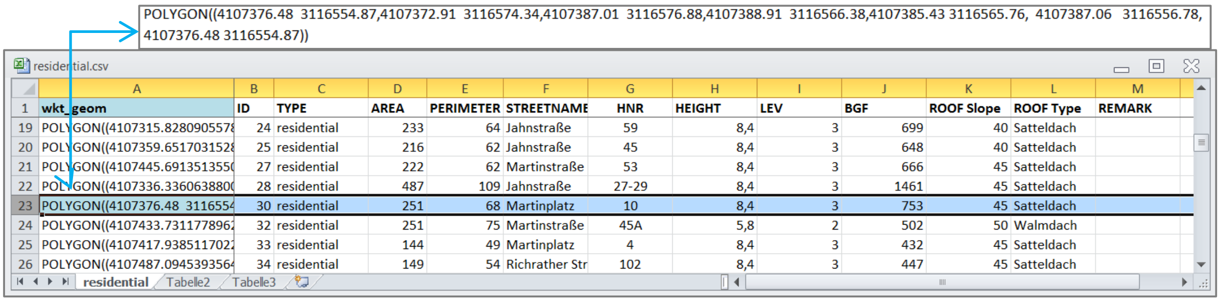
Figure 9 illustrates the result of the 3D adjustment within ArcScene. ArcScene is the ArCGIS 3D visualization application that allows one to view GIS data in three dimensions. All buildings are LOD1 objects (*Level Of Detail* – i.e. simple block models), nevertheless they give a good first impression. Additionally the layers attribute table with the calculated parameters is shown and highlighted for the special building Martinplatz 10.

Figure 9. Simple 3D visualization of buildings with Arc Scene

By using 3D representations or photos one can also find out if an attic is used as loft or for any other more or less residential use (by simple looking if it has some windows or dormer). Additionally the type of the roofs and their slope can be detected, which is also interesting for modelling a buildings behaviour.

At the end one needs to export the shapefile geometry. As *EnergyPlus* needs *Comma Separated Value (CSV*) file format as input - shapefile format is not supported - one has to convert the shapefile to CSV. The easiest way to do this without losing the additional attributes (e.g. addresses, height or building use) is to open the attribute table in QGIS and copy/paste it to excel, where one can save it as CSV file. As only residential buildings should be considered in *EnergyPlus* (under the assumption that garages have little influence through their small heights), only the 72 residential buildings were selected.

The following Figure 10 displays an example of the table after copy/paste it to Excel. As one can see compared to the attribute table shown in Figure 9, there's now a new attribute WKT_GEOM including the coordinates of the vertices of the polygon. The building Martinplatz 10 is highlighted, for better identification its WKT_GEOM value is displayed separately in the above rectangle. The first and last pair of coordinates is identical, so the building has 6 corners corresponding to the slightly L-shaped house at the upper left corner of Martinplatz (see Figure 8).



POLYGON((4107376.48 3116554.87,4107372.91 3116574.34,4107387.01 3116576.88,4107388.91 3116566.38,4107385.43 3116565.76, 4107387.06 3116556.78, 4107376.48 3116554.87))

residential.csv

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	wkt_geom	ID	TYPE	AREA	PERIMETER	STREETNAME	HNR	HEIGHT	LEV	BGF	ROOF Slope	ROOF Type	REMARK
19	POLYGON((4107315.828090557	24	residential	233	64	Jahnstraße	59	8,4	3	699	40	Satteldach	
20	POLYGON((4107359.651703152	25	residential	216	62	Jahnstraße	45	8,4	3	648	40	Satteldach	
21	POLYGON((4107445.691351355	27	residential	222	62	Martinstraße	53	8,4	3	666	45	Satteldach	
22	POLYGON((4107336.336063880	28	residential	487	109	Jahnstraße	27-29	8,4	3	1461	45	Satteldach	
23	POLYGON((4107376.48 311655	30	residential	251	68	Martinplatz	10	8,4	3	753	45	Satteldach	
24	POLYGON((4107433.731177896	32	residential	251	75	Martinstraße	45A	5,8	2	502	50	Walmdach	
25	POLYGON((4107417.938511702	33	residential	144	49	Martinplatz	4	8,4	3	432	45	Satteldach	
26	POLYGON((4107487.094539356	34	residential	149	54	Richrather Str	102	8,4	3	447	45	Satteldach	

residential Tabelle2 Tabelle3

Figure 10. CSV table as input for EnergyPlus

3.1.3 Local Heating Plant

As consumption data from the local heating plant was provided by former consortia member ENO until shortly before them exiting the project, data was available between 01.07.2013 12:45:00 and 03.02.2015 03:00:00 providing about 17 months of data, although only one complete heating season.

The CHP was well on the lines of its specification (Figure 12) of 100kW thermal and 50kW electrical, although being operated heat controlled, the upper limit of 50kW was not always reached, especially in winter (Figure 11).

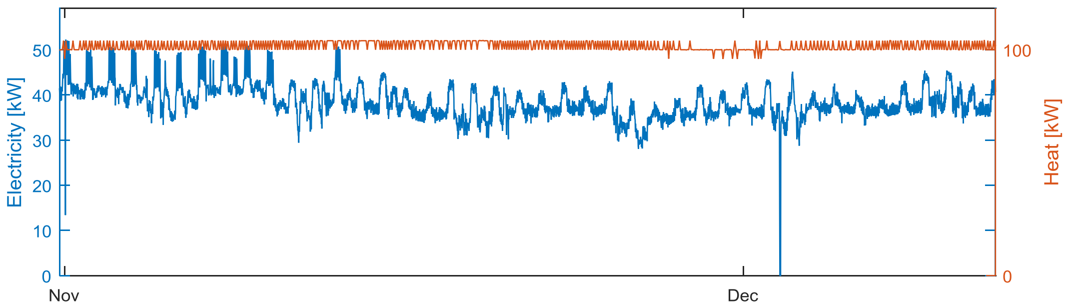


Figure 11 CHP energy production

Figure 12 CHP with rating plate

As can be seen from the overall plant data, gas power was needed especially during the heating season, but also during the transitional period (Figure 13).

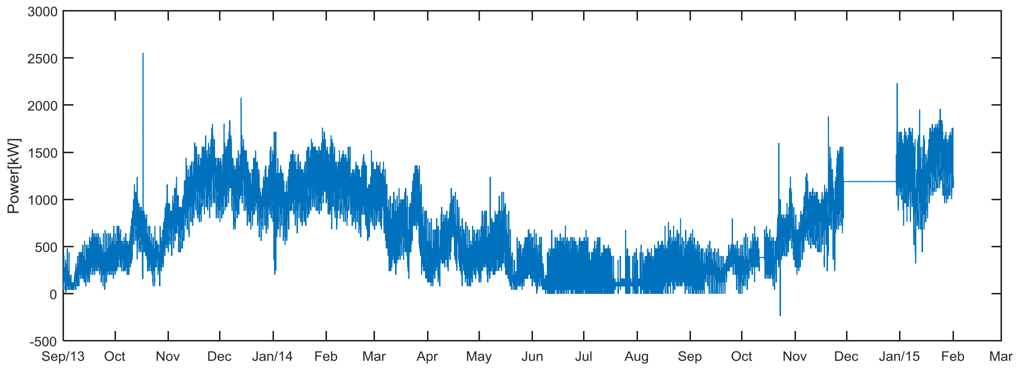


Figure 13: overall heating plant production

3.2 Modelling

3.2.1 Thermal Building Models

Based on the generated data described in chapter 3.1 above – building properties and building geometry information – it is possible to create the thermal building models as described in the modelling process in Figure 14:

- The basis for the individual building models is the geometry information specified by the ground floor dimensions and the building height.
- Taking the year of construction it is possible to get all building properties from AITs building properties database and apply those attributes to the geometric building model.
- Finally, the dynamic characteristic in terms of user behaviours based on the main usage of the building can be applied using the information from AITs building dynamic behaviour database.

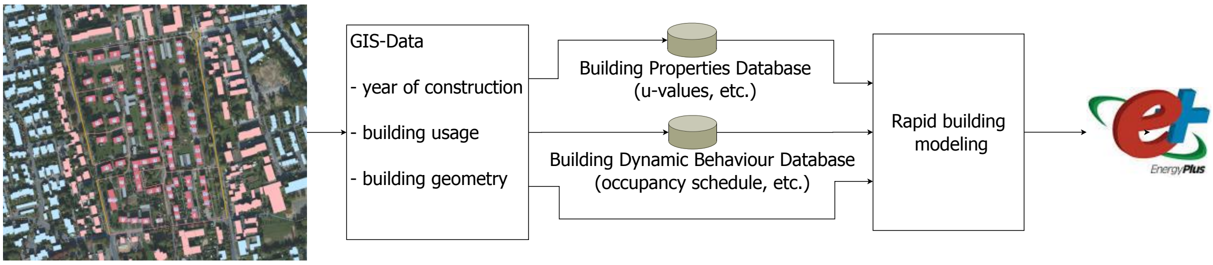


Figure 14. Overview thermal building modelling process

Combining all this information in AITs rapid building modelling process for each building of the demonstrator (Figure 15- left hand side) an individual building model was created as shows as example in Figure 15- right hand side. It takes all previously mentioned information into account in addition to the neighbourhood buildings which influence the building in terms of reducing the solar radiation due to shading effects.

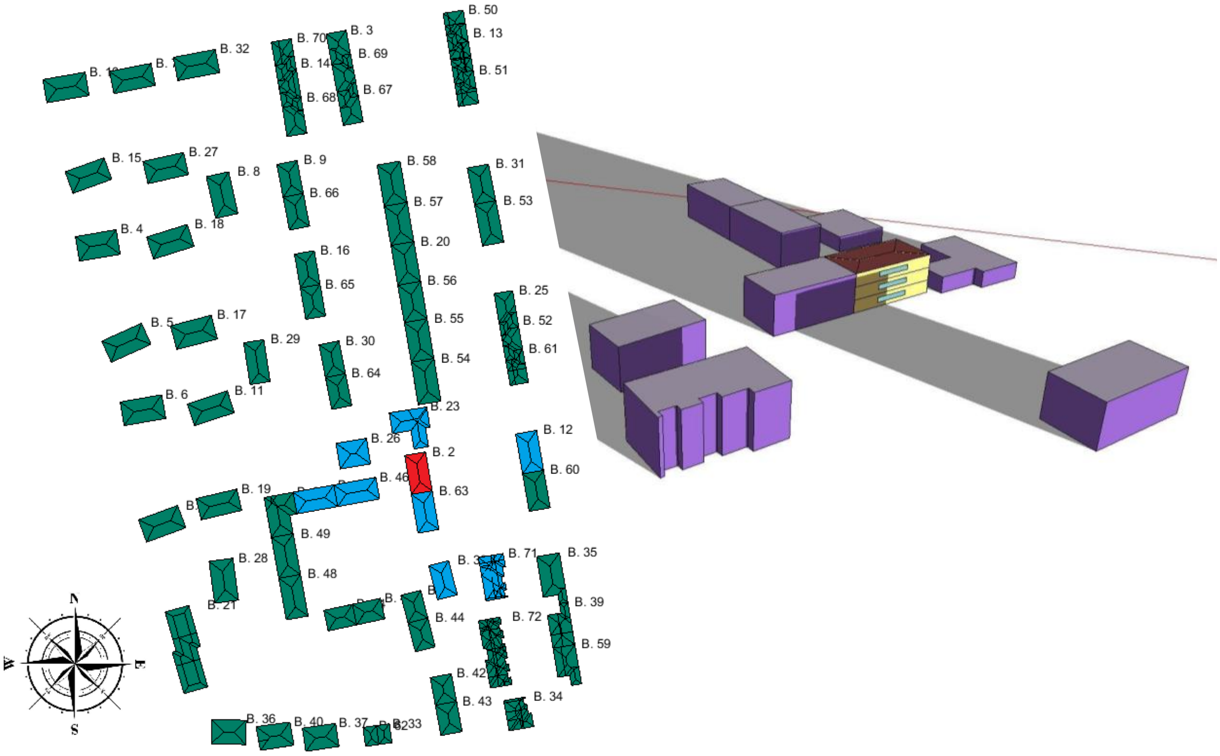


Figure 15. Example of individually model building of the demonstrator

Finally to combine each building model in a co-simulation environment every building has a Functional Mock-up Unit (FMU) implemented in order to communicate with a so called co-simulation environment. Here the building model responds to a controller regulating its average temperature as explained in detail in the following chapter.

3.2.2 Overall Building Model

To use an EnergyPlus model in an external co-simulation environment, it first must be packed-up as a Functional Mock-up Unit (FMU). For this, objects that are responsible for external data transfer are added to the EnergyPlus model.

As the co-simulation environment we will use Ptolemy II, an open source software developed by the Berkeley EECS department. The zones of an EnergyPlus FMU are regulated by a state controller which has been created directly in Ptolemy. AIT therefore developed FMU-specific actors for co-simulation which manage the interface between Ptolemy and the EnergyPlus FMU. The following Figure 16 shows such a Ptolemy model using Ptolemy's graphical interface, Vergil.

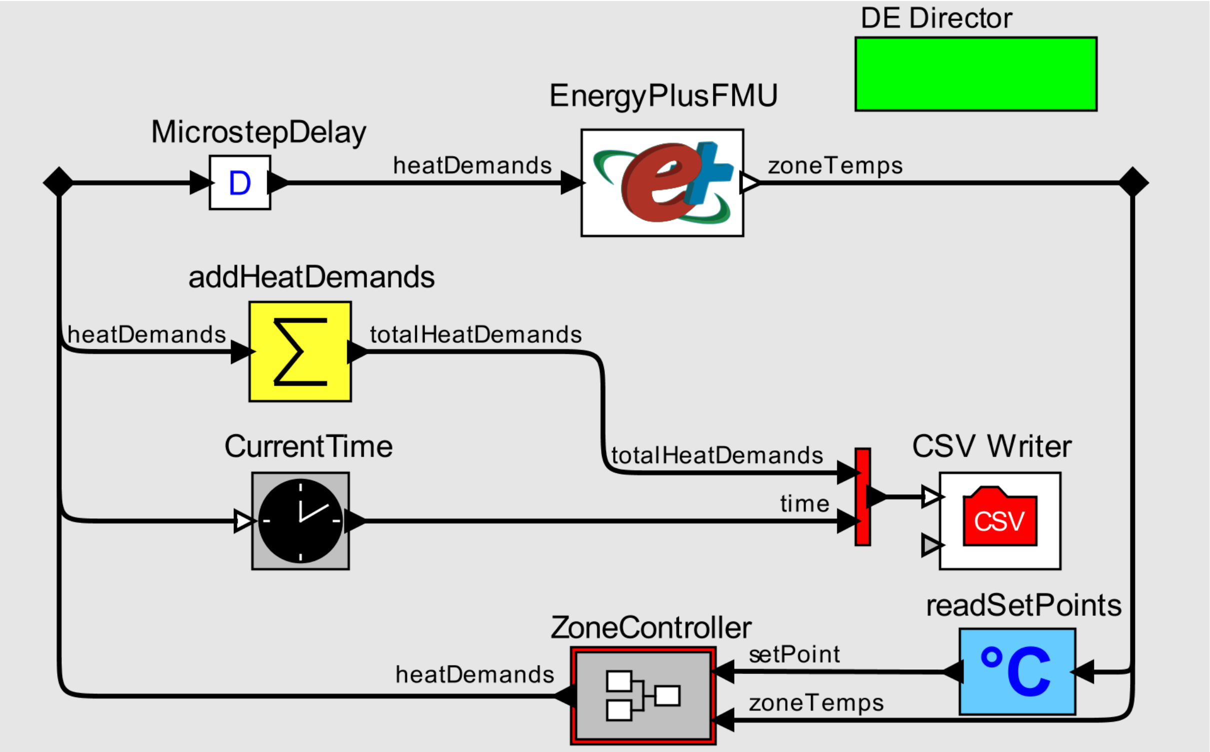


Figure 16. Integration and regulation of an EnergyPlus model within Ptolemy

The following paragraph use different formatting describing their different natures in Ptolemy. **Ptolemy actors** are highlighted in a non-serif font and bold characters, whereas Ptolemy variables are highlighted in a non-serif font only.

EnergyPlusFMU integrates the FMU of an EnergyPlus model into Ptolemy; **readSetPoints** reads a temperature set point from an external time series of ideal temperatures; **ZoneController** regulates each zone of the EnergyPlus model to be at setPoint and returns the heatDemands of all zones; these heatDemands are added by **addHeatDemands** to obtain totalHeatDemand; the current simulation time is retrieved by **CurrentTime**; totalHeatDemand and time are written into a simulation output file by **CSV Writer**. heatDemands is delayed for a few seconds by **MicrostepDelay** (to avoid a data jam) and is then passed on to **EnergyPlusFMU**. The whole loop is repeated again and again from startTime till stopTime by **DE (Discrete Event) Director**, which manages the whole simulation.

For the automatic creation of a Ptolemy model to simulate and control an EnergyPlus model, these actors, parameters, variables, relations, etc. are generated in correspondence to the total number of temperature zones and other values such as total simulation time, preferred heating rate, etc.

To simulate the thermal behaviour of a building for a given time, we change the simulation time period of its Ptolemy model accordingly. Communication between the Ptolemy model and the EnergyPlus FMU will take place every (simulated) 15 minutes, output data will be written and the controller will regulate accordingly.

4. IMPLEMENTATION

This chapter describes the combination of the individual components into a working demonstrator. The individual results can be seen in Chapter 5.

4.1 Communication and Operating Procedure

The operating procedure and communication is visualized in Figure 17 below. It shows how the simulation and EEPOS platform (in the following also called OGEMA being the underlying framework for reasons of simplicity) are combined to prove the working of the load management app.

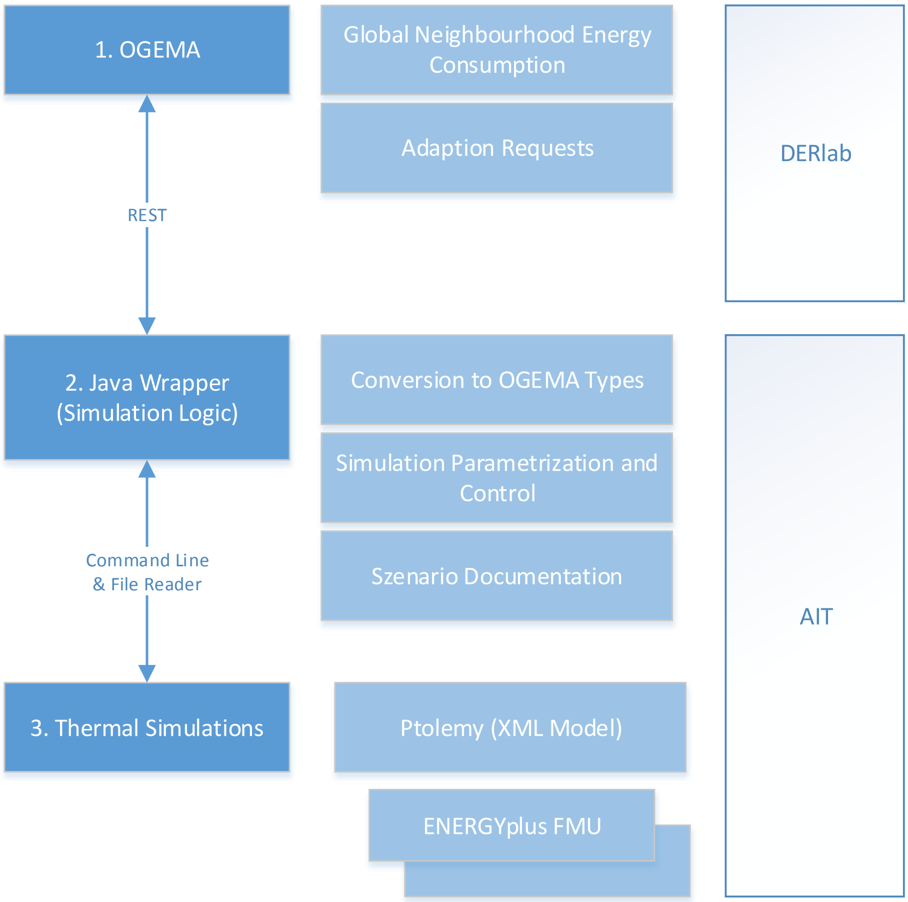


Figure 17. Operating Procedure between OGEMA and Simulation

OGEMA's (1.) role is to generate global neighbourhood energy consumption by aggregating all objects within Langenfeld. It then applies the control algorithm to generate adaption requests. Communication from and into OGEMA is performed through a REST interface.

The Java Wrapper (2.) is a component that does the management between OGEMA and the simulation. The Java Wrapper provides functionality such as conversion between OGEMA and simulation data formats (unit conversion, CSV to JSON conversion, object mapping, etc.), it sets up the simulation with the required parameters and managers the data as such that scenarios can be evaluated. Its operation needs to be separated into an initial run and consecutive runs as at first OGEMA requires a base load in order to generate subsequent adaption requests.

Initial run:

1. Generate baseline set point temperatures
2. Setup and start the simulation
3. Discard settling time and upload results to OGEMA's individual objects

Consecutive runs

1. Retrieve adaption requests for individual buildings and calculate new indoor temperature setpoint schedules.
2. Setup and start the simulation
3. Discard settling time and upload results back to OGEMA's individual objects

The thermal simulation (3.) environment (as described in chapter 4.3 below) expects input and output data to be on the file system for each individual object. As a result, the objects need to be mapped between the file system and the OGEMA resource URLs. As an example, OGEMA resources can be accessed as follows:

Heating demand schedules: http://.../rest/resources/Langenfeld1/heatMeter/mmx/forecast

Adaption request schedules: https://.../rest/resources/Langenfeld1/adaptionRequest/forecast

The simulation environment is required to keep persistence, i.e. keep last state of each building. Furthermore, simulation settling time needs to be accounted for by starting the simulation before the required time and remove the settling period in the resulting files. This overhead needs to be considered as detailed zone simulations for all objects take up to 50 minutes.

The whole setup is as such that adaptation requests are applied instantaneously through indoor temperature set point change meaning that there is no direct mechanism to negotiate or confirm the application of adaption requests. As a result, the control algorithm must not trial and error but instead requires a robust algorithm.

4.2 System Control for Demand Side Management

Due to the change of the partner from ennovatis to DERlab during the demonstration execution, DERlab provided support during the extension of OGEMA that is detailed in D4.5 in more detail. The following is a summary in respect to relevant parts of D5.4.

As mentioned before, the control algorithm generates suitable adaption requests from within OGEMA. The goal of these is to minimize the CO2 intensity of the heating system by reducing the fossil fuel fired peak boilers. Their use can be reduced by a better management of heating demands on neighbourhood level through OGEMA.

As a reminder, the following procedure is used:

1. The thermal load predictions for the individual households – providing the expected load for each household in 15 minute intervals - are pushed into OGEMA via the OGEMA REST interface.

2. The application calculates an averaged load prediction for all households, again dividing the load prediction into 15 minute intervals. Furthermore, it calculates the mean load for the span of the next 24 hours.

3. In order to ensure that most of the load shifting will focus on peak events, only a limited number of households will receive adaption requests for increasing or decreasing their thermal load at any one time. The number of households selected for this purpose during any given time interval is:

$$(Total\ number\ of\ households) \times \frac{|(Current\ Averaged\ Load) - (Mean\ Load)|}{Mean\ Load}$$

For periods when the selected number of households would exceed the total number of households (i.e. when the current averaged load for the 15 minute span is more than twice the mean load), all households are selected. Beyond that, the households which receive nonzero adaption requests are chosen randomly for each 15 minute interval.

Adaption request value	**Condition**
-2	If chosen and current averaged load > mean load
0	If not chosen for adaption requests
+2	If chosen and current averaged load <= mean load

This algorithm results in a better balanced load (towards 24h mean load) with lesser high peaks but also avoids control signals towards the lowest minimum (always -2, as this would be most CO2 saving). Also, through the random selection of buildings the algorithm is equally fair in the long run. The operation of the algorithm can be seen in Chapter 5.4 below.

4.3 Building Control

ZoneController awaits all zone temperatures and a set point temperature. zoneTemps is split by **VectorDisassembler**. Each of the 12 controllers receives one of these 12 zone temperatures and setPoint; and returns a heatDemand accordingly. All 12 heatDemands are collected by **VectorAssembler** to heatDemands.

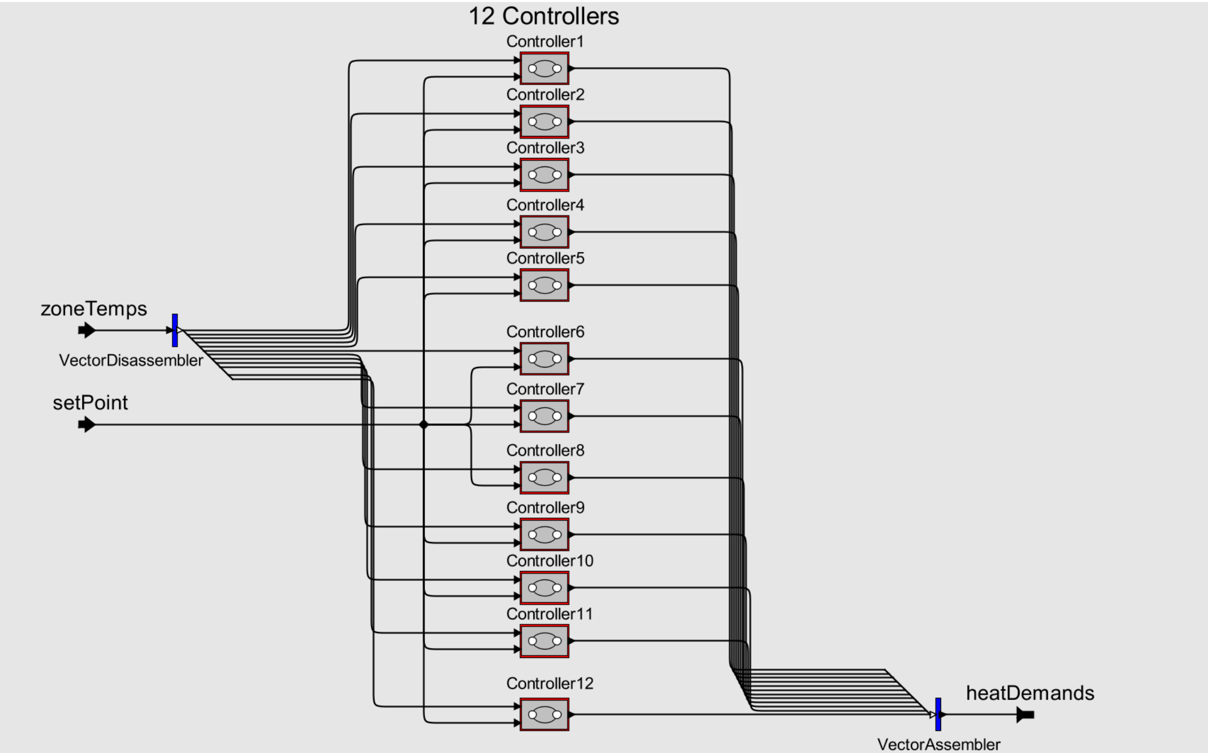


Figure 18. ZoneController composite actor

Each of the 12 controllers compares its input zoneTemp and setPoint. According to the current temperature state, is the output heatDemand then set to heatingRate or coolingRate: the heater either heats or is turned off.

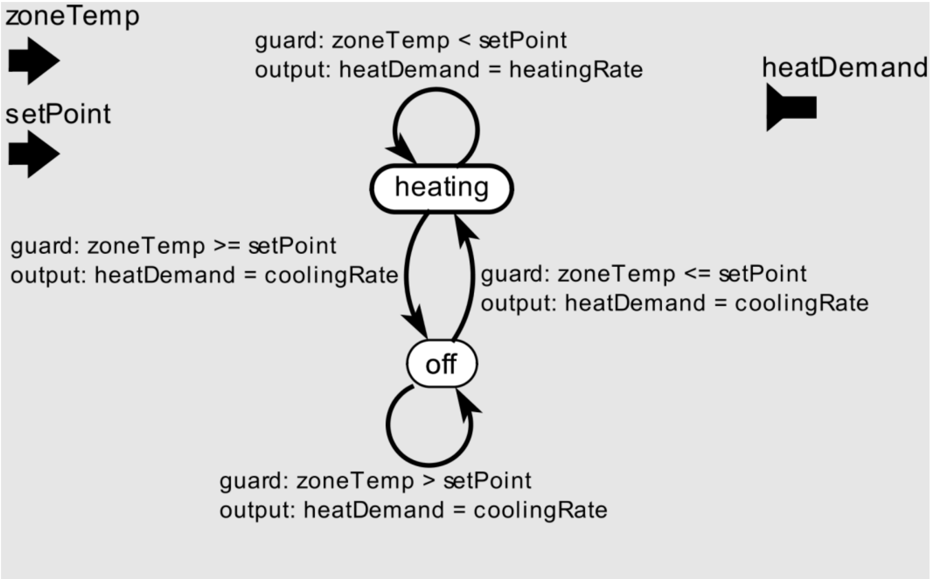


Figure 19. Controller1-Controller12

5. EXPERIMENTS AND RESULTS

This chapter shows the intermediate validations and results of the individual components orchestrated together that lead to the final conclusion in D4.5. For this the raw results of simulations are shown to show their characteristics and settling times.

Next, the basic principle of shifting loads through indoor set point changes and the effect on the heating load is shown.

5.1 Thermal Behaviour of Langenfeld Buildings

Figure 20 below is used to verify correct temperature control and reasonable simulation results. Building 21 in Langenfeld represents such an example with an indoor temperature set to a constant 22°C and a simulation duration of two weeks starting 1st of November. The different coloured lines represent a thermal zone each.

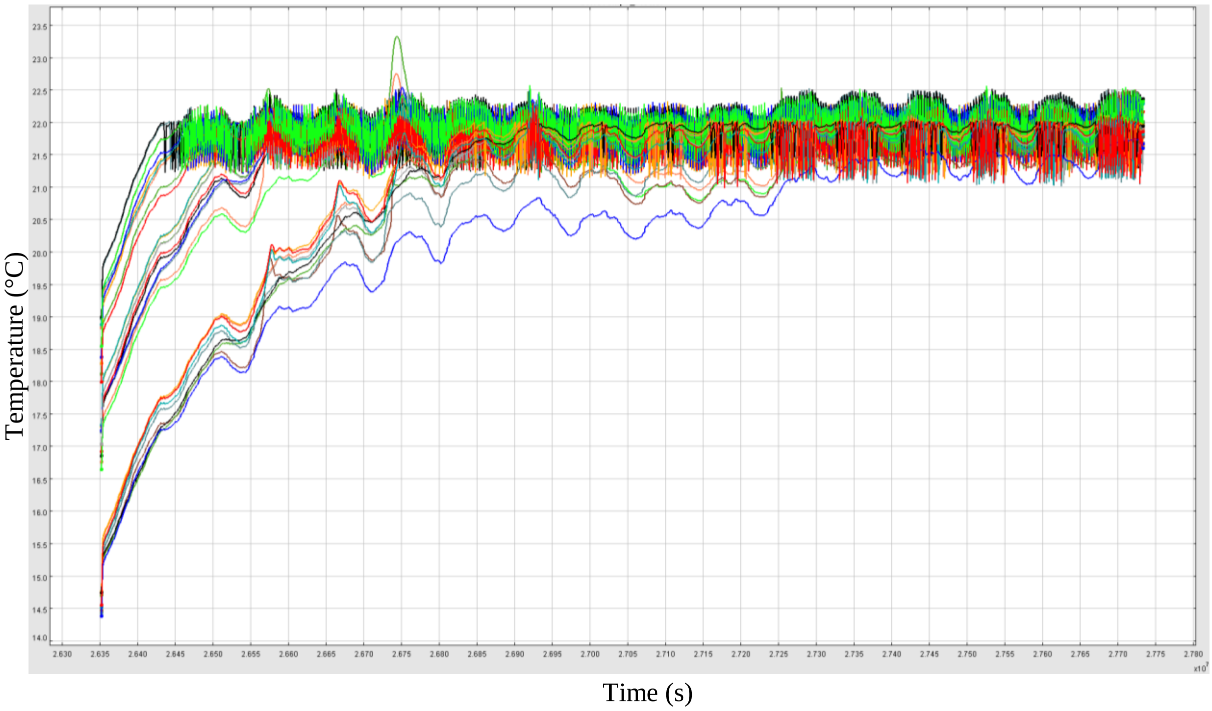


Figure 20. Temperature sequence of various zones

Most notable is the settling time of temperatures at the beginning of the sequence. This is characteristic to the thermal simulations as the internal variables need to settle first. The time required can be seen to be 26350000-27250000=900000s, which results in 11 days of settling time at least. For further simulations the settling time is set to be 14 days which is a compromise of simulation time and reliable results. This settling time is removed from simulation results, except for the neighbourhood baseline in Chapter 5.2.

It can also be seen that, depending on the zone there is a temperature fluctuations between 0.5K and 1K with little overshooting above 22°C. Also the temperature difference between zones is of similar magnitude indicating proper operation of the zone controls.

5.2 Neighbourhood Baseline Heating Demand

Next it was looked at the total heating demand of the neighbourhood which consists of the sum of all individual objects in Figure 21. It is represents a baseline as no adaption requests are applied at this stage (meaning that all setpoints are set to be a constant 22°C).

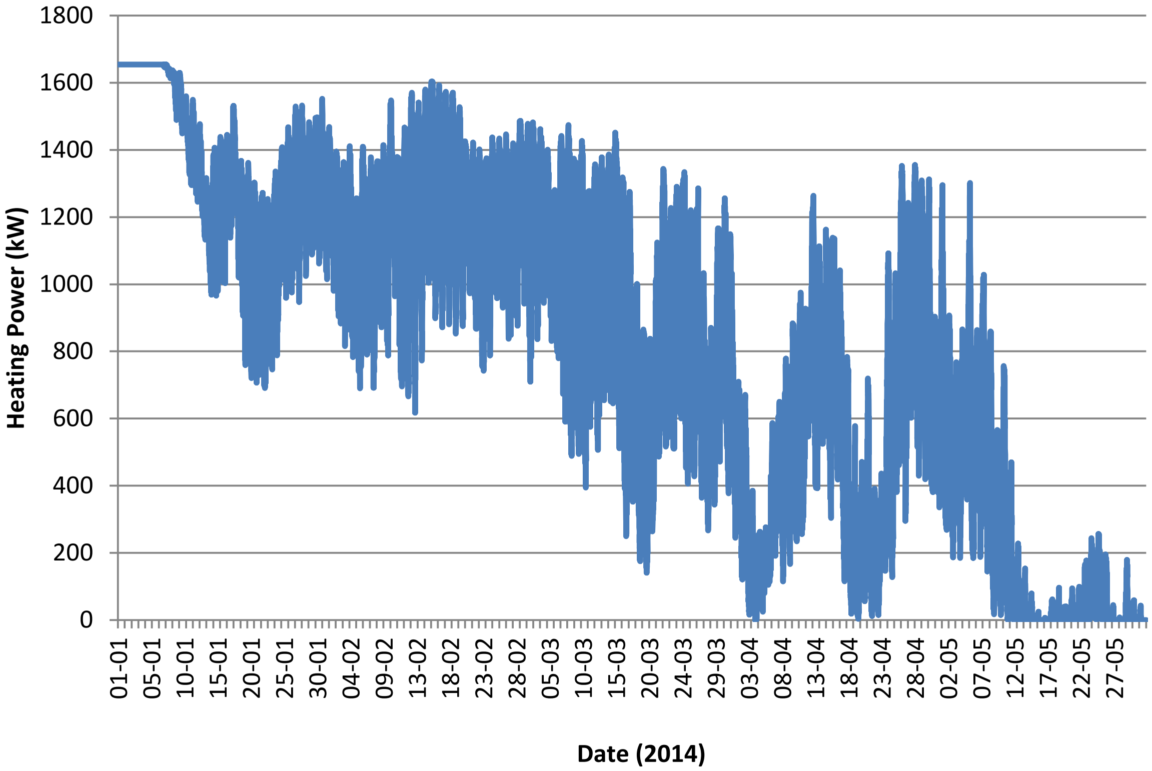


Figure 21. Total simulated heating demand in winter time

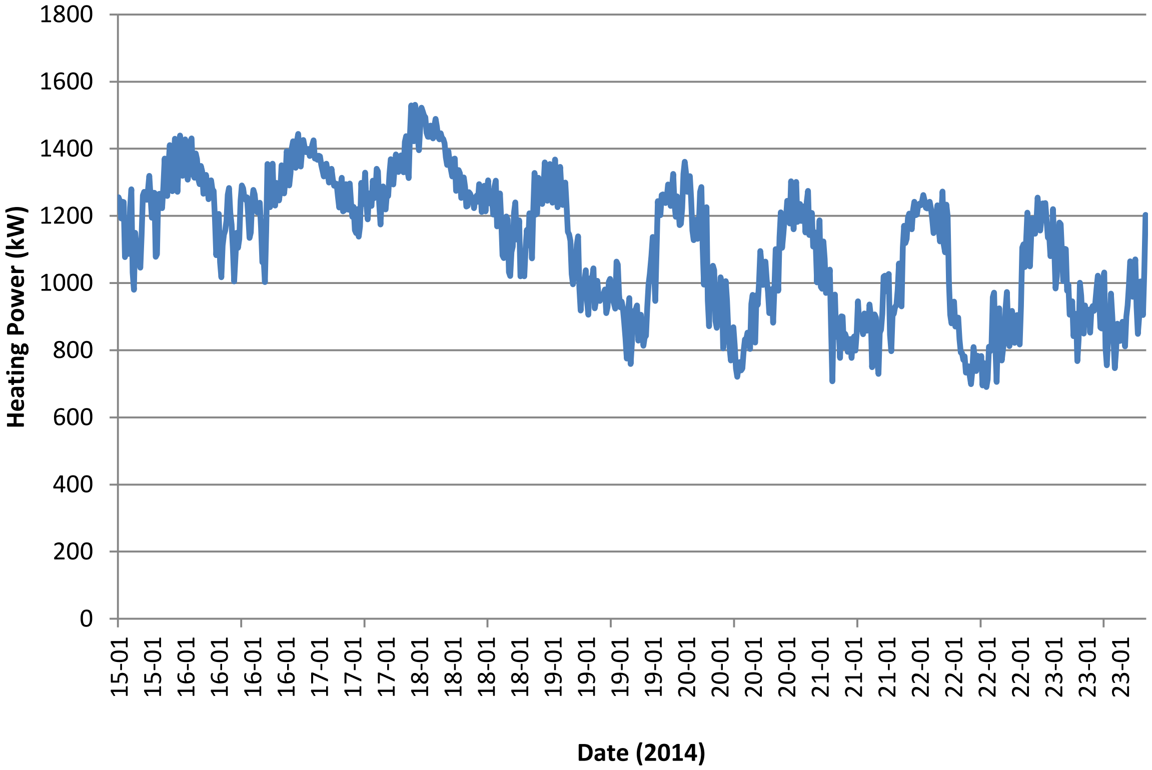


Figure 22. Total simulated heating demand one week in winter time

Figure 21 shows a varying heating demand between 0 in summer time and up to 1655kW in peak load in wintertime (especially during settling time at the beginning of the simulation at full load). Those values are well within installed capacity. Daily patterns can be seen clearly when looking more closely at one week in Figure 22 with peaks in the morning and evening times.

However, it must be noted that those figures, being the sum of all individual objects, do not take into account heat distribution losses in the district network. Also, having no demand in summertime shows that domestic hot water consumption (DHW) has been neglected in the simulations. Hot water consumption is expected to create further peaks in the morning and evening hours where management would have been effective. However, DHW management cannot be modelled through the room setpoint change approach as was done with the heating system. This needs further investigation in updated model controls.

5.3 Temperature Shifting Potential

After having looked at baseline scenarios where no management has occurred, this chapter will look at the effects of applying room temperature setpoint management.

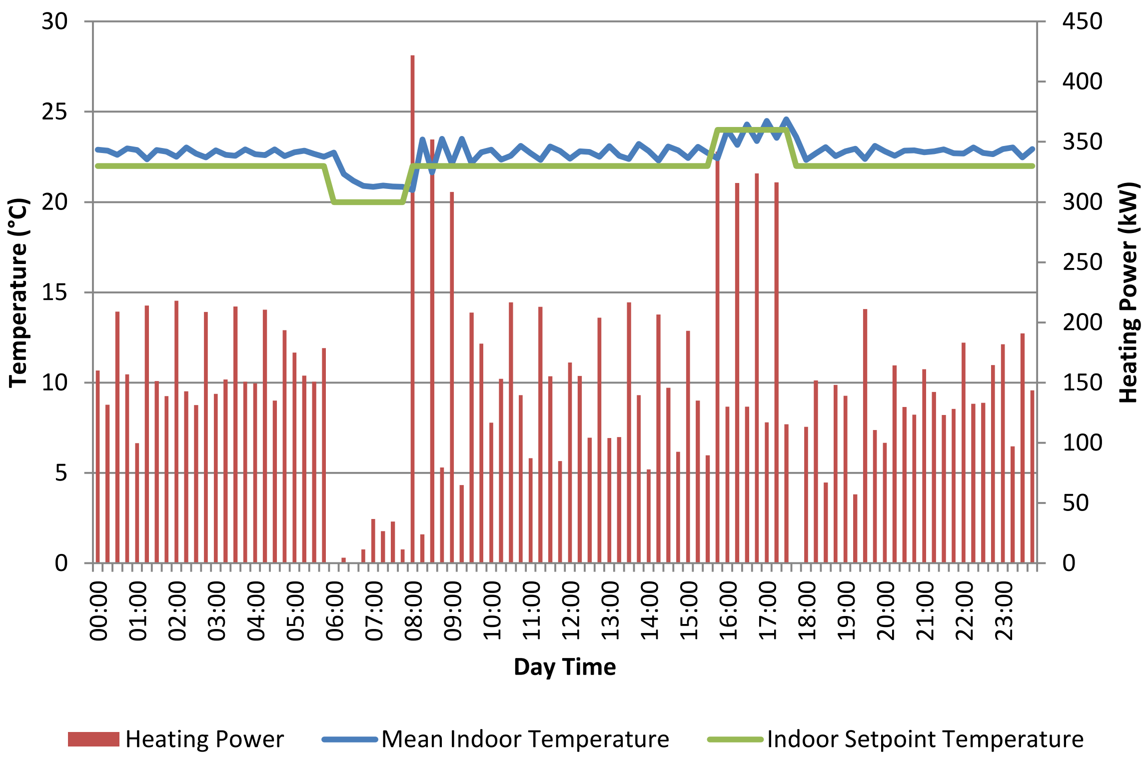


Figure 23. Exemplarily plot of shifting load away from the morning

The example in Figure 23 above shows the effect of performing a thermal load shift away from the morning times and into the evening times for the exemplary object 1. The energy consumption is expected to be very similar in comparison to no load shifting as the setpoint changes are balanced. It can be seen that reducing the load starting at 6 o'clock in the morning has the effect of a much reduced load for most zones until 8 o'clock where there is an appropriate rebound. Without adaption requests (setpoint changes) the indoor temperatures move around 22°C as expected. This concludes a thermal load shifting potential for the individual object.

5.4 Analysis of Generated Setpoints

Moving from an individual building to the entire neighbourhood results in adaption requests as seen in Figure 24 below. They are best visualized in a heat map where blue dots indicate a reduction request, purple no change and red a request to shift into this period. Objects 1…72 refer to the individual buildings in Langenfeld. The plot was done for a day with clearly visible adaption requests (1st of February).

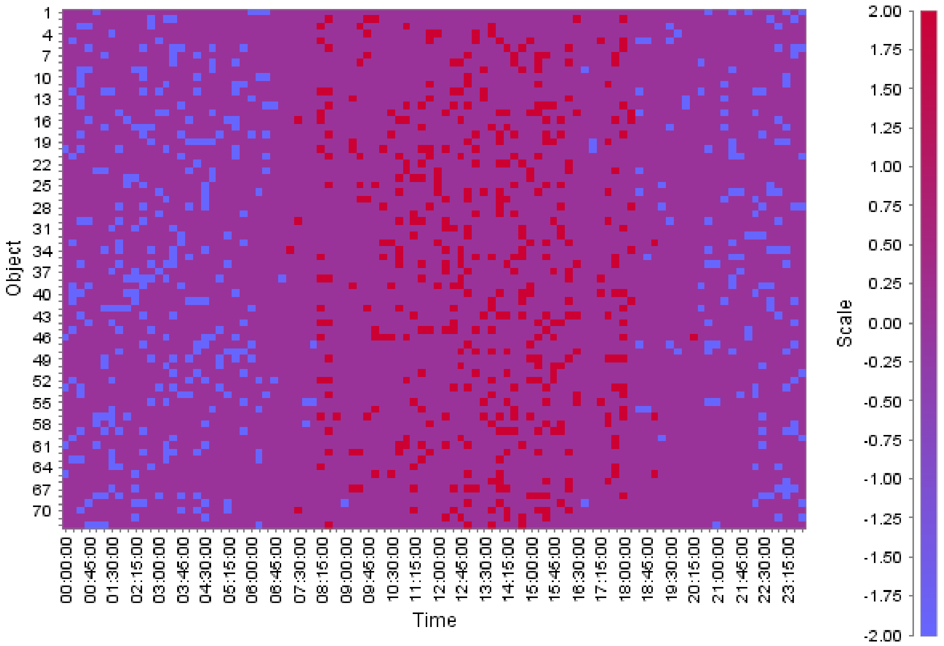


Figure 24. Adaption requests Langenfeld

Adaption requests are equally distributed across buildings and show a clear pattern of moving load from the early hours into mid-day and away from the evening hours. Furthermore, there are no contradictory requests of both shifting into and away a time period at the same time. Analysis shows also that every building is asked to participate.

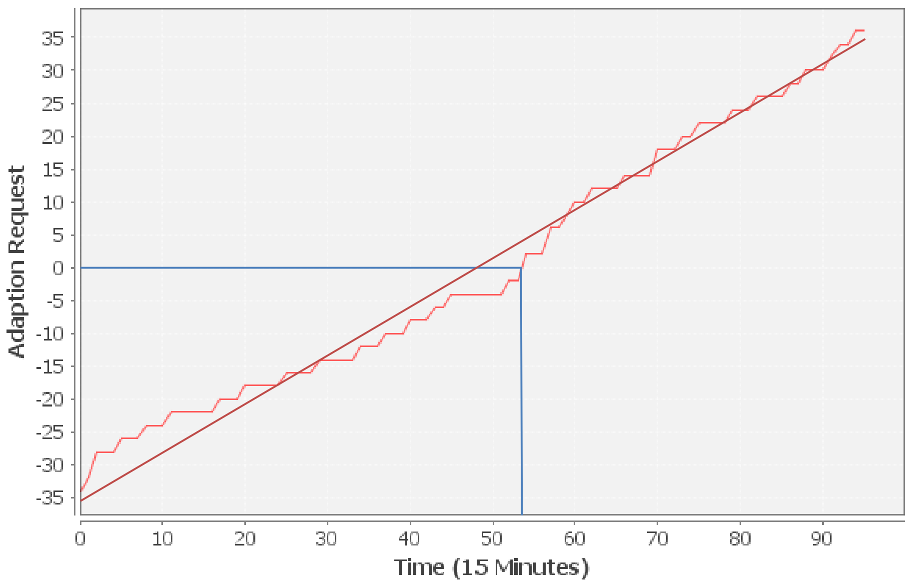


Figure 25. Sorted plot of adaption requests

However, the is a slight surplus of negative adaption requests as seen in Figure 25 where the time with negative adaption requests is slightly (1 hour) longer than the time with positive adaption requests. This is also seen with the mean values of the heating rate on neighbourhood scale with and without adaption requests (AR) in Figure 26. This imbalance results in a slight energy saving of 4.6% on that day although no energy saving was envisaged. It is also to take notice, that the isolated view on this single day cannot be linearly scaled to a sequence of days as the heating capacity on the next day is strongly affected by the prior day.

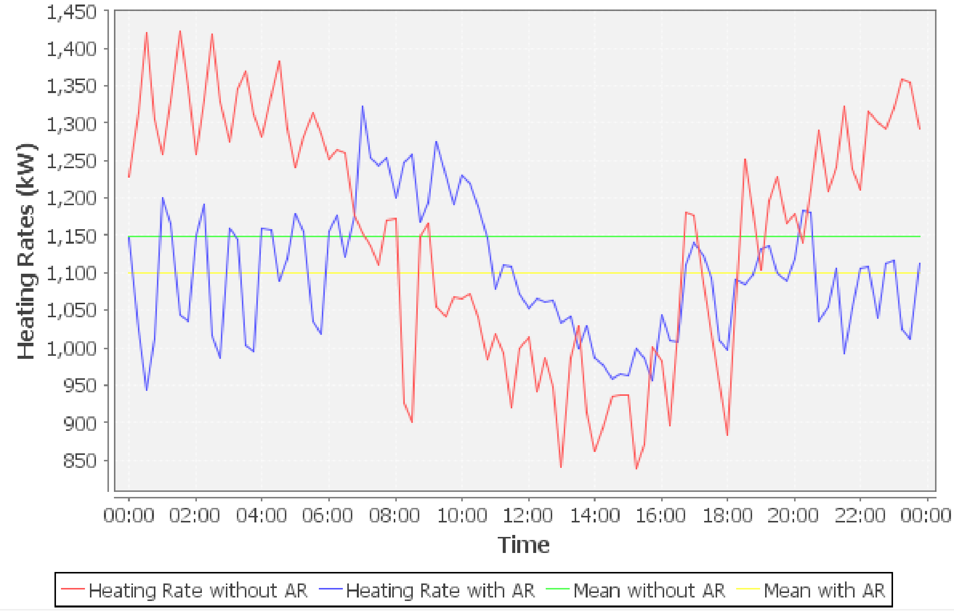


Figure 26: Effect of adaption requests on heating rate

5.5 Conclusion

The acquired data and models (Chapter 3) allowed the running of reasonable dynamic thermal simulations that highlight the theoretic possibilities of utilizing thermal flexibility of homes. Furthermore, the combination of OGEMA / EEPOS platform together with the simulation environment (Chapter 4.1) shows effective operation (Chapter 5.3) and can be the baseline for further investigations. The control algorithm generated equally distributed values to effectively shift loads towards the daily mean (thus reducing peak loads) as analysis (Chapter 5.4) showed.

The demonstrator successfully displayed the mode of operation and positive effect of neighbourhood energy management in a simulated environment. Detailed CO2 savings and peak load reductions can be seen D4.5.

6. References

[1] http://www.qgis.org/en/site/

[2] http://desktop.arcgis.com/en/

[3] http://wiki.openstreetmap.org/wiki/Main_Page

[4] http://extract.bbbike.org/

[5] http://nominatim.openstreetmap.org

[6] http://www.citygmlwiki.org/index.php/Main_Page

[7] https://www.google.de/maps/