

SimStadt crash course



Camilo León-Sánchez, Giorgio Agugiaro



Last update: 18 September 2024

License

This presentation is licensed under the [Creative Commons License CC BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/). According to CC BY-NC-SA 4.0 permission is granted to share this document, i.e. copy and redistribute the material in any medium or format, and to adapt it, i.e. remix, transform, and build upon the material under the following conditions:



- **Attribution:** You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **NonCommercial:** You may not use the material for commercial purposes.
- **ShareAlike:** If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- **No additional restrictions:** You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

SimStadt in a nutshell

- SimStadt is a 3D urban energy system modelling platform developed by HFT Stuttgart
 - <https://simstadt.hft-stuttgart.de>
- It has a Graphical User Interface (GUI) and consists of different workflow steps that use information from
 - An (XML-encoded) CityGML-based 3D city model
 - Building physics and usage libraries
 - Weather data
- Typical urban applications:
 - Building energy demand (heating, cooling, DHW production)
 - Building refurbishment scenarios
 - Photovoltaic potential
 - Scenarios for adoption of renewables

Introduction

Installation

Fundamentals

Input

Output

SimStadt Workflows

References

SimStadt in a nutshell

Introduction

Installation

Fundamentals

Input

Output

SimStadt Workflows

References

- The building physics library is based on the work of the Institut für Wohnen und Umwelt ([IWU](#)) (Weiler et al., 2019). Buildings are classified according to:
 - Type (e.g. single-family house, terraced house, etc.)
 - Period of construction
- For each building type and period, there is a building *archetype* with its respective wall, roof and window physical parameters. SimStadt associates these parameters with the actual building geometry (i.e. thermal envelope)
- SimStadt requires INSEL. INSEL is a software for simulation, monitoring and visualization of energy systems. It is the underlying engine of multiple workflow steps
 - It has some predefined simulation models included, but it is also possible to design your own model extensions or entirely new models

Installation: software requirements

Currently, SimStadt runs best on a **Windows** platform, however it can be used on Linux (e.g. Ubuntu) and (still experimental) macOS, too

- Some of the underlying modules are not supported in other operating systems, e.g. the Simplified Radiosity Algorithm, which is required for shadow calculations

Software requirements

- 1) Java 8 64bit (JDK or JRE) with JavaFX libraries – or equivalent
- 2) INSEL (current version: 8)
- 3) SimStadt
- 4) Simplified Radiosity Algorithm

Additional (optional) resources

- KIT ModelViewer
- 3D City Database (with Energy ADE plugin) for data retrieval and storage
- FME for ETL operations

Short installation guide

Please note: This is a short version of the detailed guide available at

<https://simstadt.hft-stuttgart.de/getting-started/install-software>

Introduction

Installation

Fundamentals

Input

Output

SimStadt Workflows

References

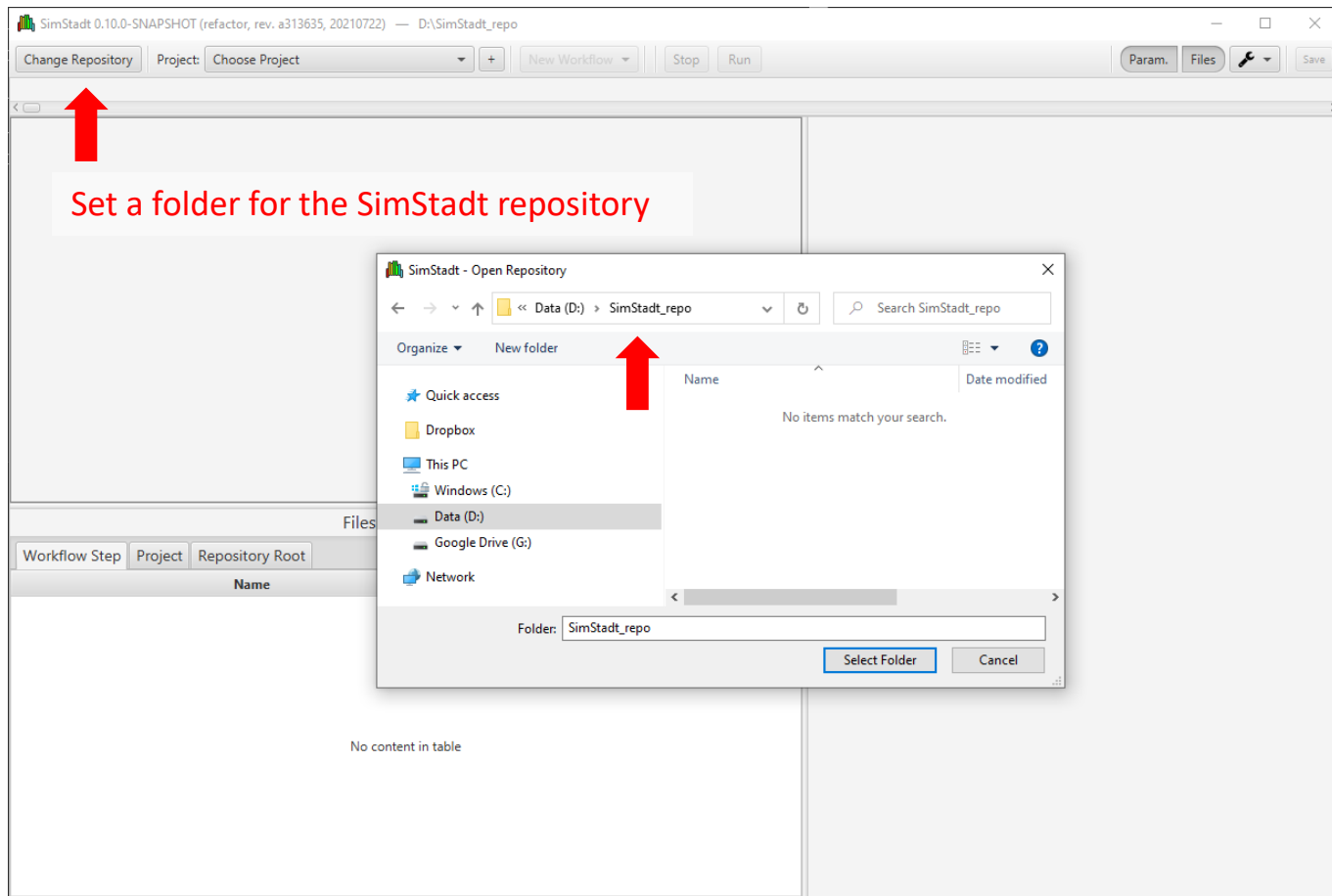
Installation steps

- 1) Download and install **Java**
- 2) Download and install **INSEL**
 - It will be installed by default in C:\Program Files (x86)\Insel 8.3
- 3) Download and replace INSEL file **inselBS.dll**
 - The file to replace is in folder C:\Program Files (x86)\Insel 8.3\resources
- 4) Download and unzip **SimStadt**
 - E.g. into C:\Program Files\SimStadt
- 5) Download and unzip the **Simplified Radiosity Algorithm (SRA)**
 - Extract to C:\Program Files\ so that C:\Program Files\SimplifiedRadiosity\shortwave_integer.exe is an executable
 - SRA executable should then be found by the IrradianceProcessor in SimStadt when SRA_Perez is selected
- 6) Optional, but recommended: download and install Gnuplot
 - Needed for the Visualisation step in SimStadt

First run

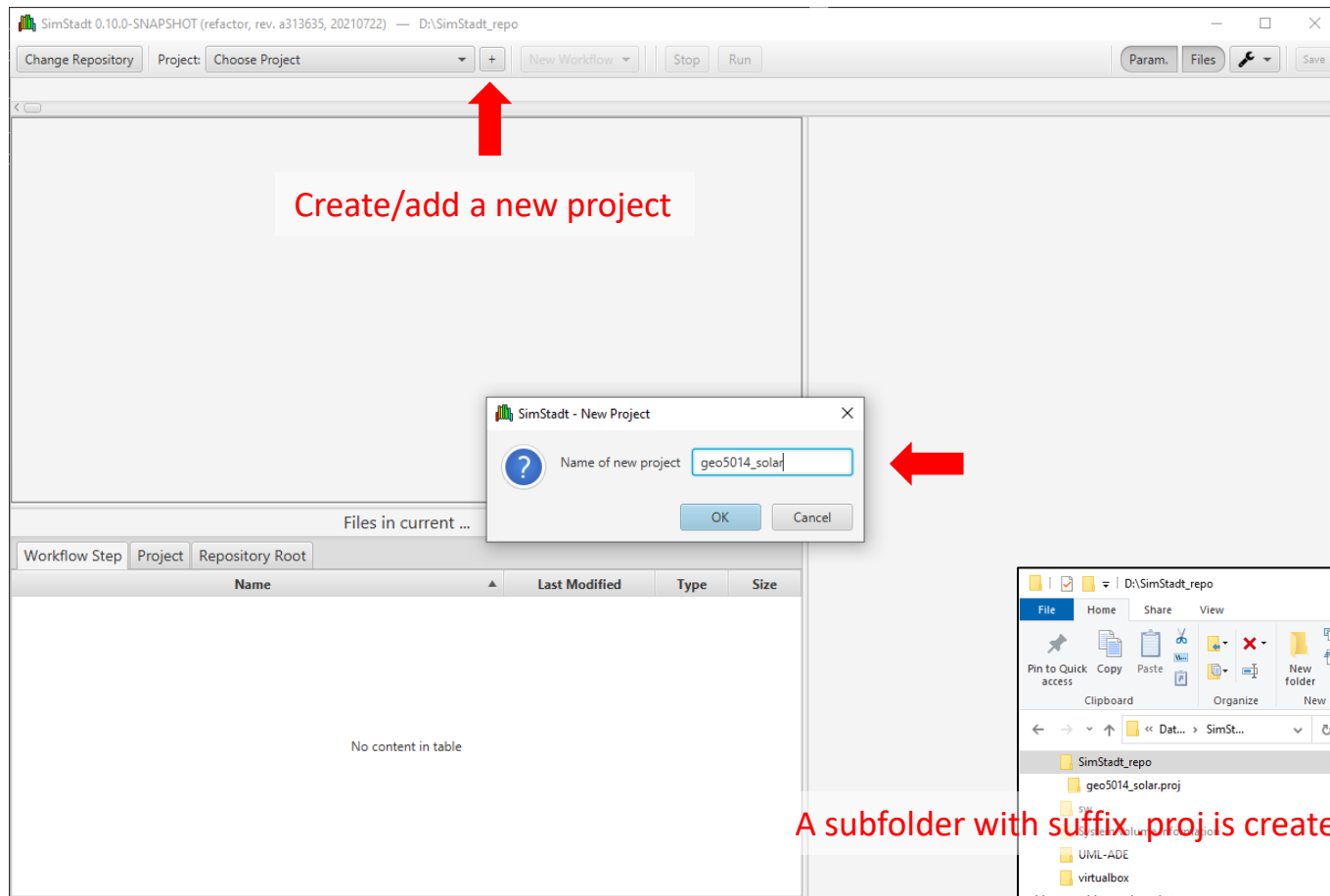
- Once you are done installing the software, start SimStadt by double-clicking on **SimStadt.bat** in the SimStadt installation folder
 - You better create a shortcut to the Desktop or to the Start menu! 😊
- Upon first run, you are requested to define a folder for the *repository*
 - A repository in SimStadt is the root folder in which you collect and save all your projects and CityGML files
 - Check that you have writing permission in that folder!
 - You can immediately create a new project, e.g. *test_solar*
 - A subfolder with suffix .proj is created in the repository folder for each project
- Follow the instructions given in <https://simstadt.hft-stuttgart.de/getting-started/first-run/>
- The next slides contain an excerpt of the information available online in the SimStadt documentation webpage

First run



The screenshot shows the SimStadt 0.10.0-SNAPSHOT interface. At the top, there is a toolbar with buttons for 'Change Repository', 'Project: Choose Project', 'New Workflow', 'Stop', 'Run', 'Param.', 'Files', and 'Save'. Below the toolbar, a red arrow points to a text box that says 'Set a folder for the SimStadt repository'. In the foreground, a 'SimStadt - Open Repository' dialog box is open, showing a file explorer view of the 'Data (D:) > SimStadt_repo' directory. A red arrow points to the 'New folder' button in the dialog. The dialog also shows a search bar, a list of locations (Quick access, Dropbox, This PC, Windows (C:), Data (D:), Google Drive (G:), Network), and a 'Folder:' field containing 'SimStadt_repo'. At the bottom of the dialog are 'Select Folder' and 'Cancel' buttons.

First run



The screenshot shows the SimStadt 0.10.0-SNAPSHOT interface. The main window has a toolbar with buttons for 'Change Repository', 'Project: Choose Project', '+', 'New Workflow', 'Stop', 'Run', 'Param.', 'Files', and 'Save'. A red arrow points to the '+' button. A text box in the center says 'Create/add a new project'. A 'SimStadt - New Project' dialog box is open, with the text 'Name of new project' and the input field containing 'geo5014_solar'. A red arrow points to the dialog box. Below the dialog box is a table with columns 'Name', 'Last Modified', 'Type', and 'Size', and the text 'No content in table'. A file explorer window is open at the bottom right, showing the directory 'D:\SimStadt_repo' with a subfolder 'geo5014_solar.proj' selected. A red arrow points to this subfolder. A text box at the bottom right says 'A subfolder with suffix .proj is created'.

SimStadt 0.10.0-SNAPSHOT (refactor, rev. a313635, 20210722) — D:\SimStadt_repo

Change Repository Project: Choose Project + New Workflow Stop Run Param. Files Save

Create/add a new project

SimStadt - New Project

Name of new project geo5014_solar

OK Cancel

Files in current ...

Name	Last Modified	Type	Size
No content in table			

Workflow Step Project Repository Root

D:\SimStadt_repo

File Home Share View

Pin to Quick access Copy Paste New folder Properties Select

Clipboard Organize Open

Search SimStadt_repo

SimStadt_repo

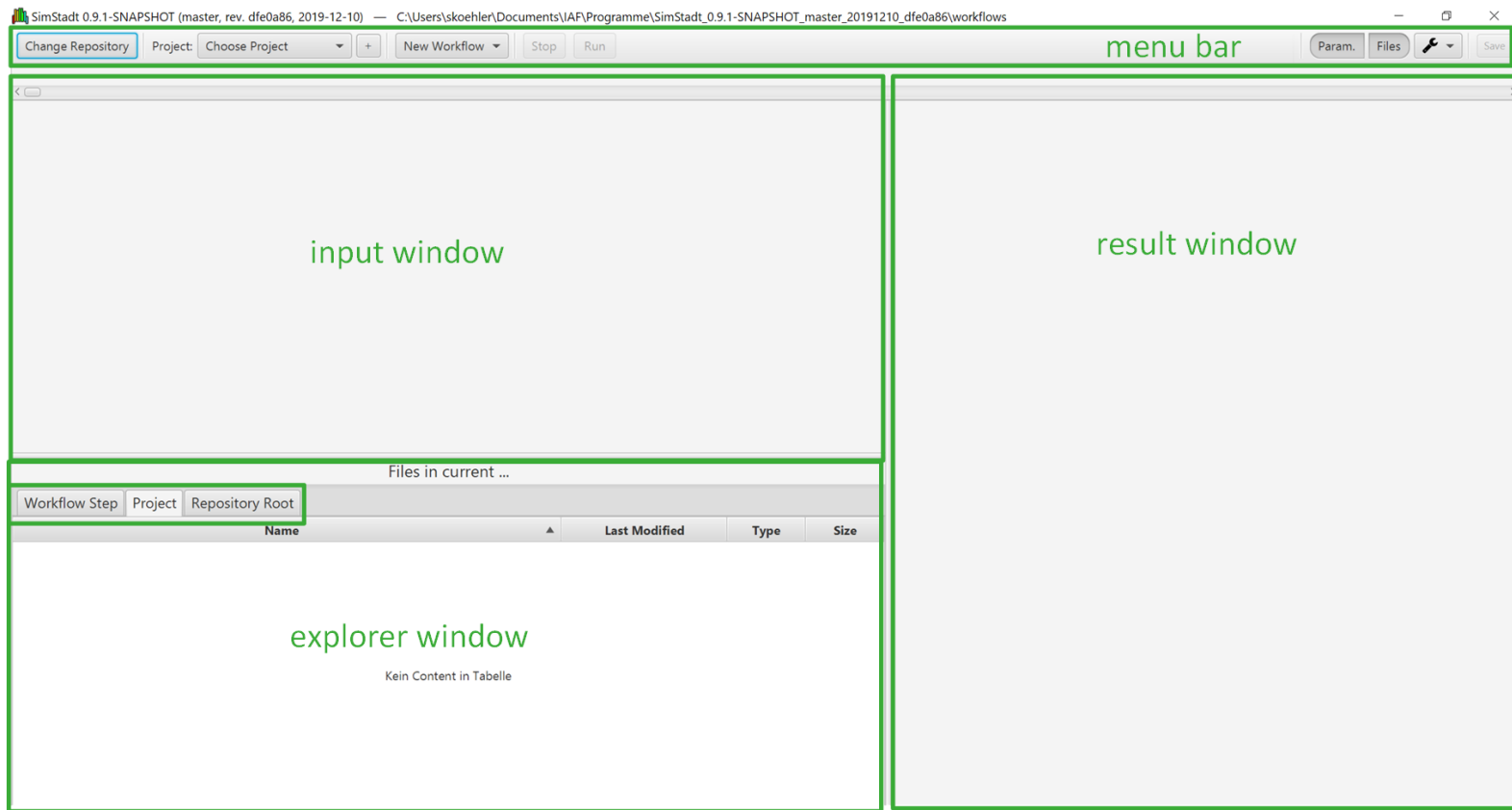
- geo5014_solar.proj
- SW
- System Columns and Properties
- UML-ADE
- virtualbox

1 item 1 item selected

A subfolder with suffix .proj is created

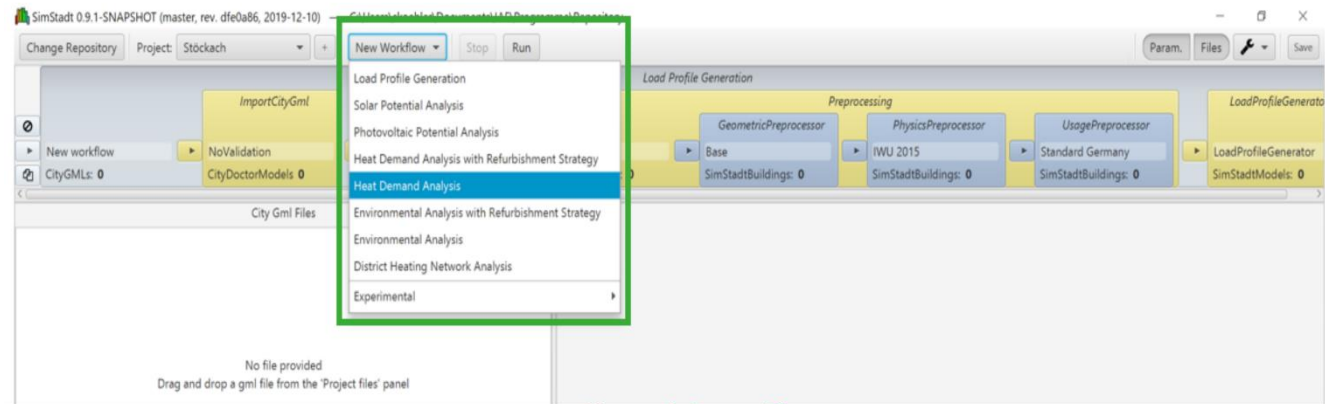
GUI and functionalities

- Introduction
- Installation
- Fundamentals**
- Input
- Output
- SimStadt Workflows
- References

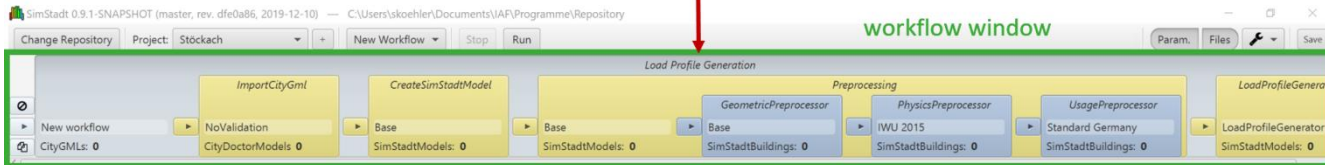


SimStadt GUI

- Introduction
- Installation
- Fundamentals**
- Input
- Output
- SimStadt Workflows
- References



Name of the workflow

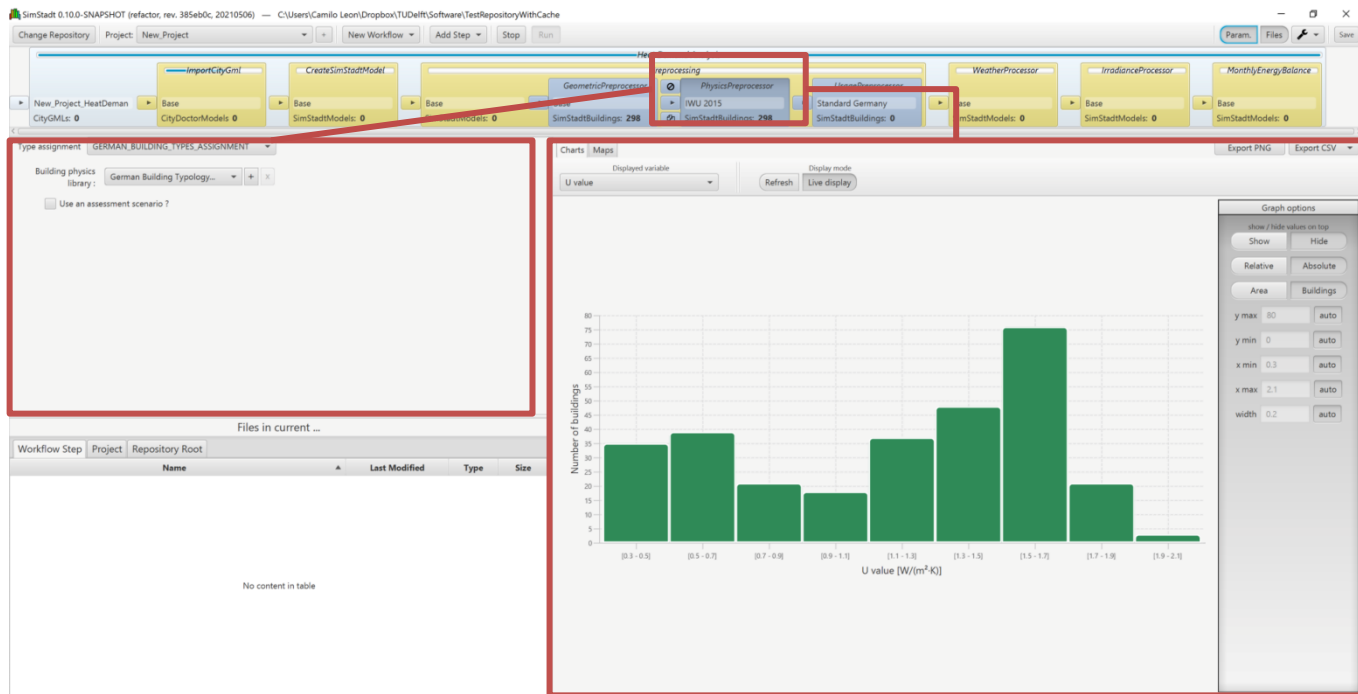


workflow window

Workflow sub-step (blue)

Workflow step (yellow)

Visualization of the workflow



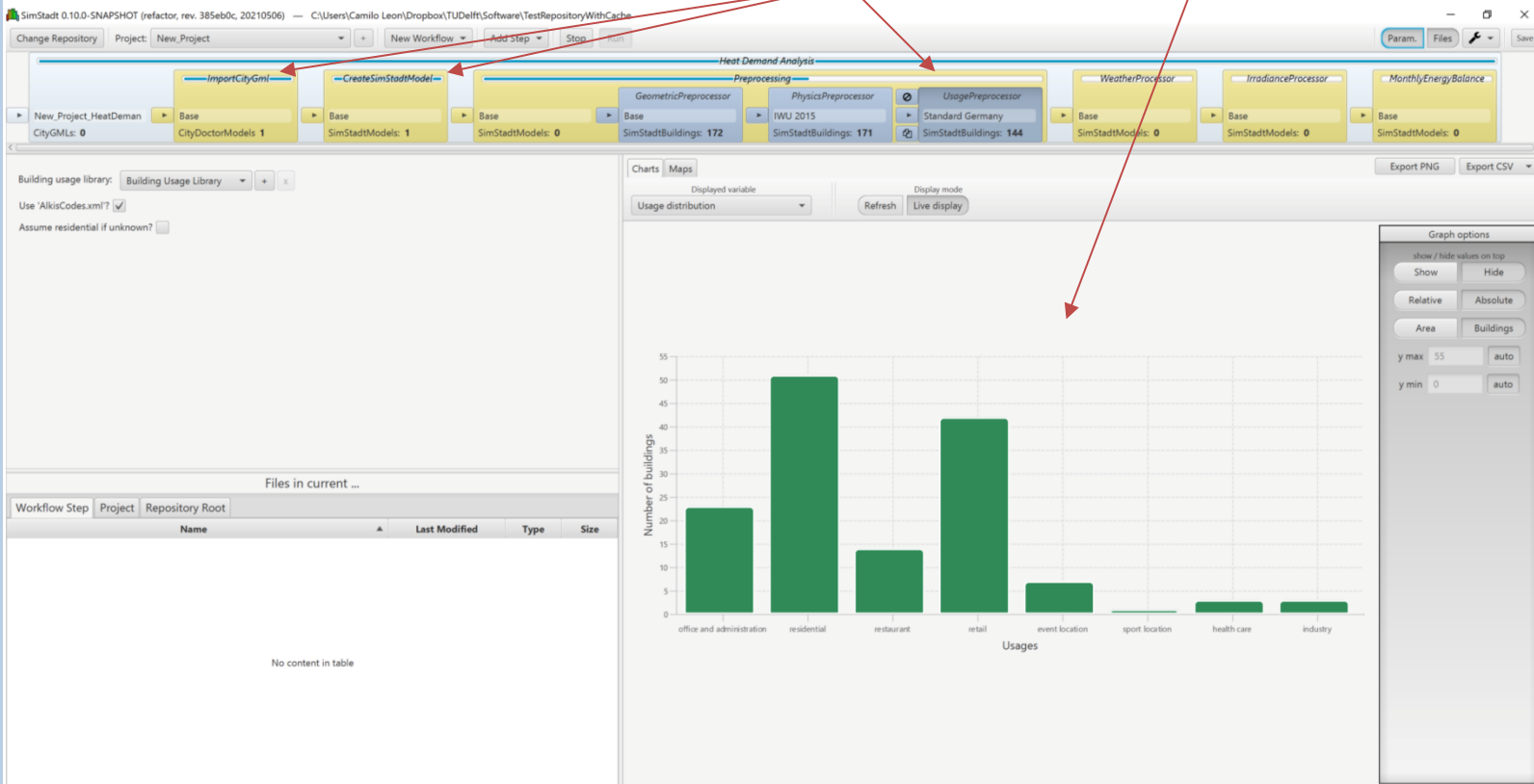
The screenshot shows the SimStadt GUI interface. At the top, there is a workflow editor with several steps: ImportCityGml, CreateSimStadtModel, GeometricPreprocessor, PhysicsPreprocessor (selected), UrbanDesignPreprocessor, WeatherProcessor, IrradianceProcessor, and MonthlyEnergyBalance. Below the workflow editor, there are two main panels. The left panel is titled 'Type assignment' and shows 'GERMAN_BUILDING_TYPES_ASSIGNMENT' with a 'Building physics library' set to 'German Building Typology...'. The right panel is titled 'Charts' and shows a bar chart of 'Number of buildings' versus 'U value [W/m²K]'. The chart has a 'Refresh' button and a 'Live display' button. The chart data is as follows:

U value [W/m²K]	Number of buildings
[0.3 - 0.5]	35
[0.5 - 0.7]	40
[0.7 - 0.9]	25
[0.9 - 1.1]	20
[1.1 - 1.3]	35
[1.3 - 1.5]	45
[1.5 - 1.7]	75
[1.7 - 1.9]	25
[1.9 - 2.1]	5

The GUI changes depending on the selected step

Progress bars

Results



The screenshot displays the SimStadt GUI interface. At the top, a workflow titled "Heat Demand Analysis" is shown with several steps: ImportCityGml, CreateSimStadtModel, GeometricPreprocessor, PhysicsPreprocessor, UsagePreprocessor, WeatherProcessor, IrradianceProcessor, and MonthlyEnergyBalance. Each step has a progress bar and a status indicator. Below the workflow, there are settings for building usage library, AlkisCodes.xml, and a checkbox for "Assume residential if unknown?".

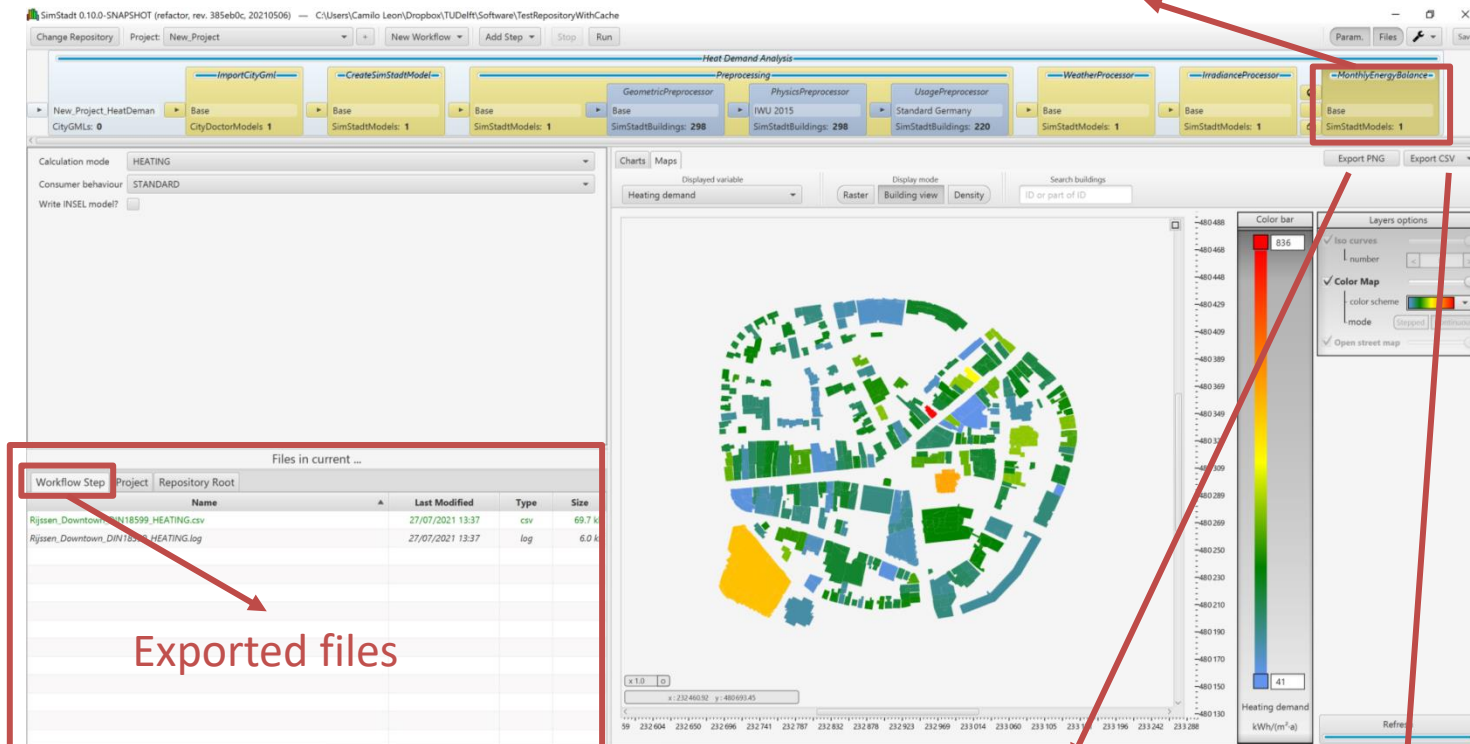
At the bottom right, a bar chart titled "Usage distribution" shows the number of buildings for various usages. The y-axis is labeled "Number of buildings" and ranges from 0 to 55. The x-axis is labeled "Usages" and includes categories: office and administration, residential, restaurant, retail, event location, sport location, health care, and industry. The chart shows that residential buildings are the most numerous, followed by retail.

Usage	Number of buildings
office and administration	23
residential	51
restaurant	14
retail	42
event location	7
sport location	1
health care	3
industry	3

SimStadt GUI

- Introduction
- Installation
- Fundamentals**
- Input
- Output
- SimStadt Workflows
- References

Monthly Energy Balance step



The screenshot shows the SimStadt GUI interface. At the top, a workflow bar includes steps like 'ImportCityGml', 'CreateSimStadtModel', 'Preprocessing', 'WeatherProcessor', 'IrradianceProcessor', and 'MonthlyEnergyBalance'. The 'MonthlyEnergyBalance' step is highlighted with a red box. Below the workflow, a map displays heating demand for a city area, with a color bar on the right ranging from 41 to 836 kWh/m²a. A file explorer window is open, showing a table of files in the current directory.

Workflow Step	Project	Repository Root	Name	Last Modified	Type	Size
			Rijsen_Downtown_DIN18599_HEATING.csv	27/07/2021 13:37	csv	69.7 k
			Rijsen_Downtown_DIN18599_HEATING.log	27/07/2021 13:37	log	6.0 k

Exported files

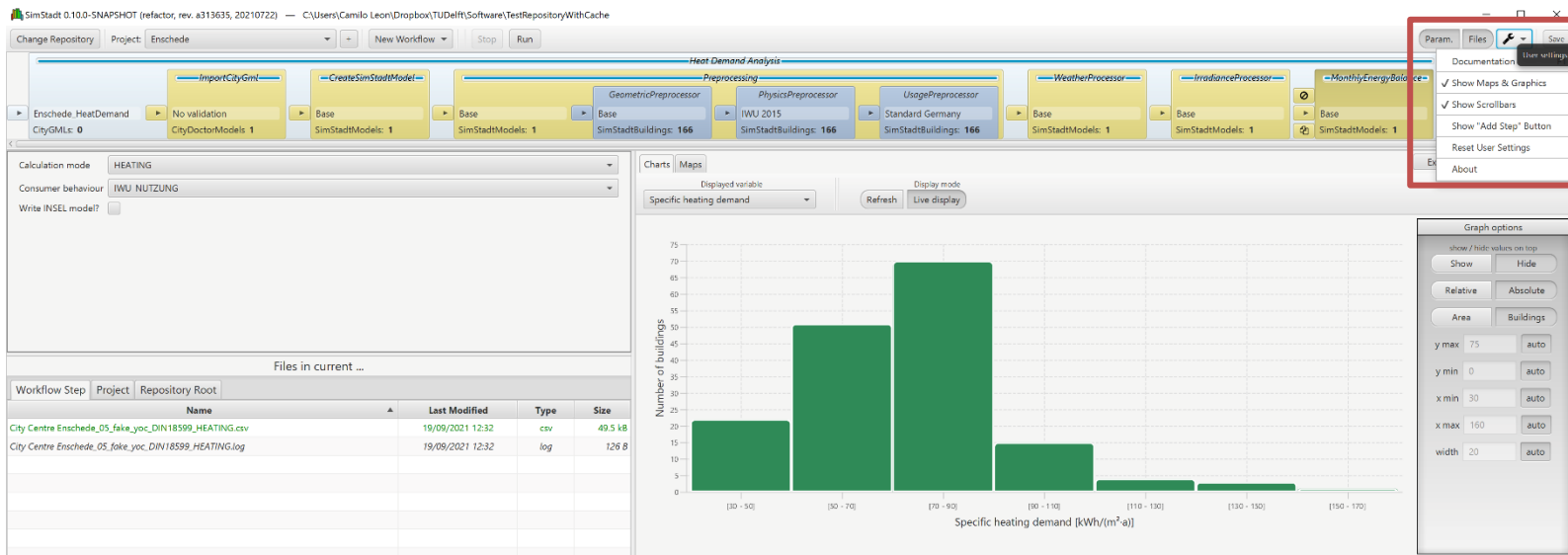
Export results as PNG

Export results as CSV

The best way to access an exported file is from the GUI

Workflow customisation

- It is possible to customise a workflow by adding additional steps or processors
- The user must enable the Show “add step” button



The screenshot displays the SimStadt software interface. The main window shows a workflow titled "Heat Demand Analysis" with several steps: ImportCityGml, CreateSimStadtModel, GeometricPreprocessor, PhysicsPreprocessor, UsagePreprocessor, WeatherProcessor, IrradianceProcessor, and MonthlyEnergyBalance. A red box highlights the "Param." menu, which includes the option "Show 'Add Step' Button".

Below the workflow, there is a "Files in current ..." table:

Workflow Step	Project	Repository Root	Name	Last Modified	Type	Size
			City Centre Enschede_05_fake_yoc_DIN18599_HEATING.csv	19/09/2021 12:32	csv	49.5 kB
			City Centre Enschede_05_fake_yoc_DIN18599_HEATING.log	19/09/2021 12:32	log	126 B

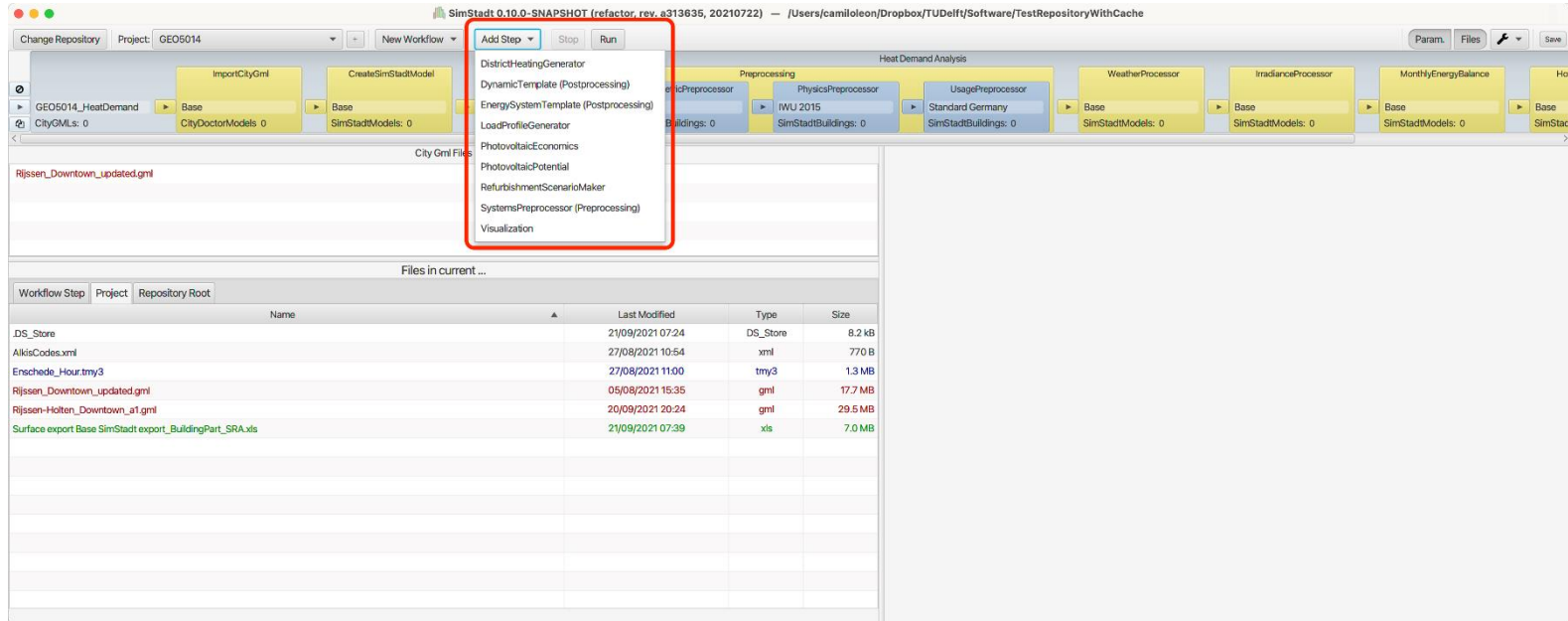
On the right, a bar chart shows the "Number of buildings" versus "Specific heating demand [kWh/(m²·a)]". The x-axis ranges from [50 - 60] to [150 - 170]. The y-axis ranges from 0 to 75. The bars represent the number of buildings in each specific heating demand bin.

Graph options:

- show / hide values on top: Show Hide
- Relative Absolute
- Area Buildings
- y max: 75 auto
- y min: 0 auto
- x min: 30 auto
- x max: 160 auto
- width: 20 auto

Workflow customisation

- It is possible to customise a workflow by adding additional steps or processors:
- The user must enable the Show “add step” button
 - The available processor depends on the active workflow



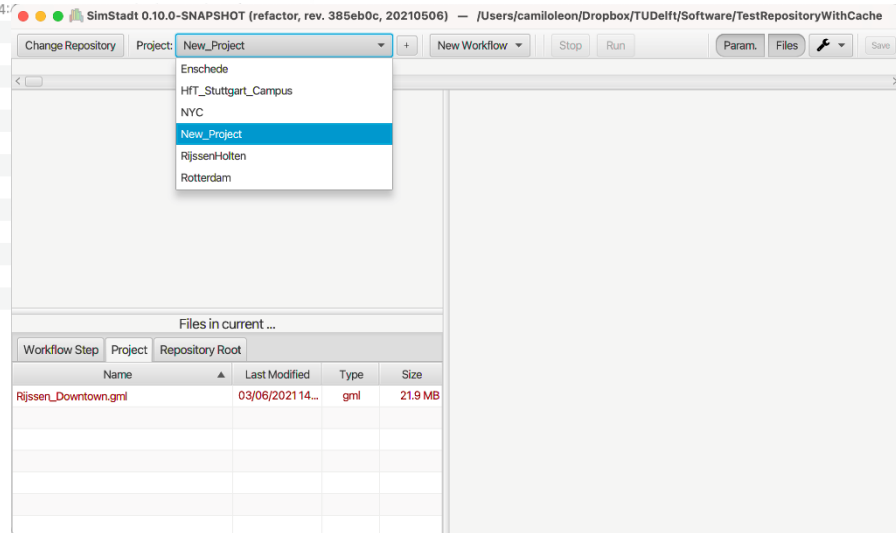
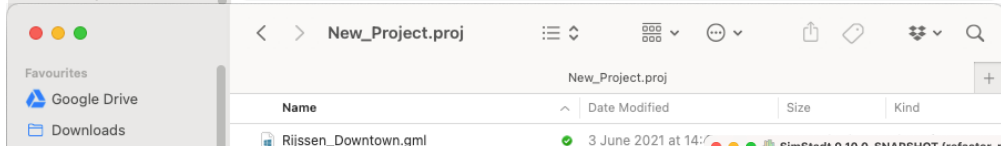
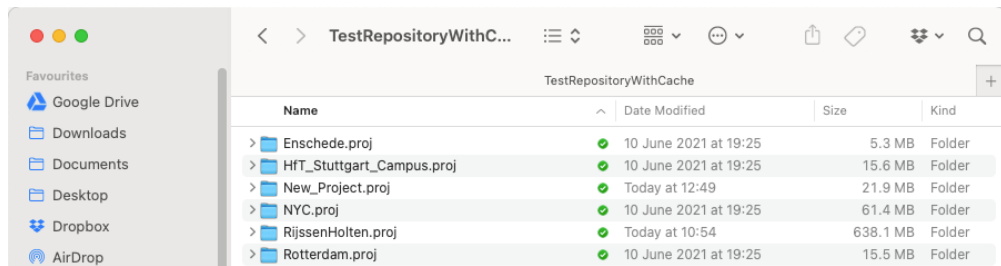
The screenshot shows the SimStadt 0.10.0-SNAPSHOT (refactor, rev. a313635, 20210722) interface. The main window displays a workflow graph for 'Heat Demand Analysis'. A red box highlights the 'Add Step' dropdown menu, which lists the following options:

- District-Heating-Generator
- Dynamic-Template (Postprocessing)
- Energy/System-Template (Postprocessing)
- Load-Profile-Generator
- Photovoltaic-Economics
- Photovoltaic-Potential
- Refurbishment-Scenario-Maker
- Systems-Preprocessor (Preprocessing)
- Visualization

Below the workflow graph, there is a table titled 'Files in current ...' with the following data:

Workflow Step	Project	Repository Root	Name	Last Modified	Type	Size
			DS_Store	21/09/2021 07:24	DS_Store	8.2 kB
			AlkisCodes.xml	27/08/2021 10:54	xml	770 B
			Enschede_Hour.tmy3	27/08/2021 11:00	tmy3	1.3 MB
			Rijssen_Downtown_updated.gml	05/08/2021 15:35	gml	17.7 MB
			Rijssen-Holten_Downtown_at.gml	20/09/2021 20:24	gml	29.5 MB
			Surface export Base SimStadt export_BuildingPart_SRA.xls	21/09/2021 07:39	xls	7.0 MB

Data repository structure

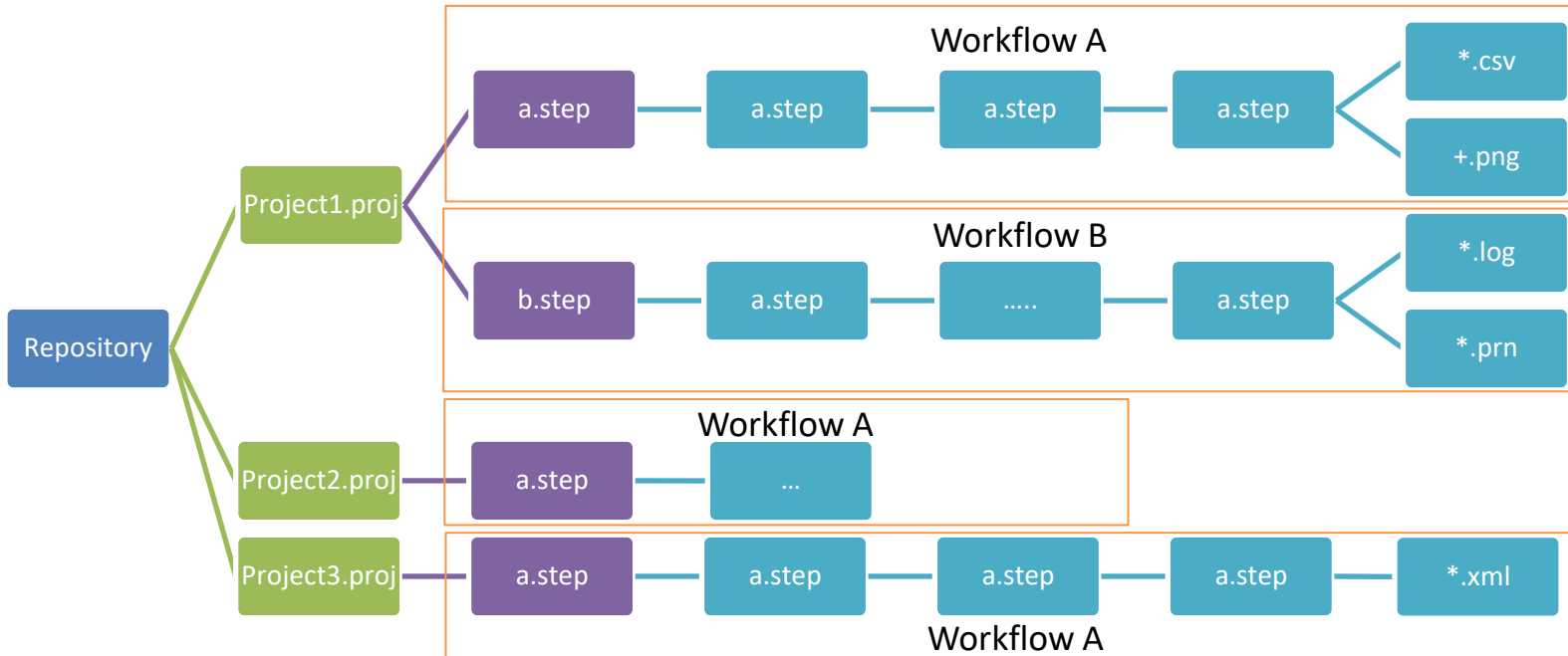


A *repository* in SimStadt is the root folder in which you collect and save all your projects.

All necessary input files (CityGML, weather data, etc.) must be copied within the folder of each project to be used from SimStadt.

Data repository structure

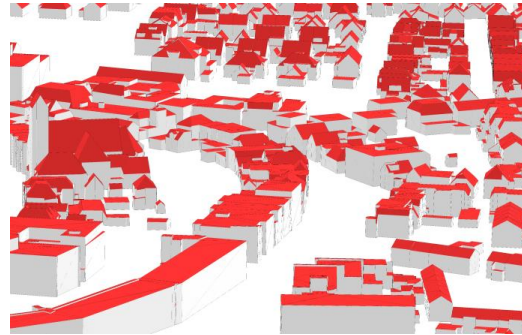
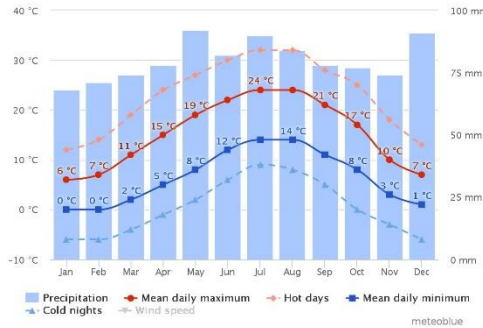
Due to its complexity, it is better to access files from the GUI



Input data requirements

The required input data depend – obviously – on the simulation purpose.
In general, typically input data are:

- 1) **Weather data**
- 2) **A 3D city model**, containing building geometries and some attributes
- 3) **Libraries** containing sets of parameters (e.g. for building physics, occupancy, etc.)



		BUILDING SIZE CLASS			
		SINGLE FAMILY HOUSES	TERRACED HOUSES	MULTI-FAMILY HOUSES	APARTMENT BLOCKS
BUILDING AGE CLASS	Middle Climatic Zone				
	1 Up to 1900				
	2 1901–1920				
	3 1921–1945				
	4 1946–1960				
	5 1961–1975				
	6 1976–1990				
	7 1991–2005				
8 After 2005					

Image source: https://mb-richtexteditor.s3-eu-west-1.amazonaws.com/20151014095814_chart-8.jpg

Input data: weather

Weather data (3 options)

- a) INSEL offline database
- b) Photovoltaic Geographical Information System (PVGIS) online database
- c) Tmy3 file (hourly/monthly). Mandatory parameters:

- Location
- *GHI*, "Global Horizontal Irradiance"
- *DNI*, "Direct Normal Irradiance"
- *DHI*, "Diffuse Horizontal Irradiance"
- *Dry-bulb*, "Dry-bulb ambient temperature"

- Energy Plus (EnergyPlus, 2021)
- NSRDB: National Solar Radiation Database (NREL, 2021)

Input data: weather

- Weather data is stored in a **tmy3** file. It is a simple csv file!
- First line:
 - siteCode(integer), locationName (string), , 1, lat (in dec. deg.), long (in dec. deg.), elevation (in m a.s.l.)
 - You can adapt the needed columns, all others can be renamed to NaN and will be ignored
- Encoding rules: <https://www.nrel.gov/docs/fy08osti/43156.pdf>

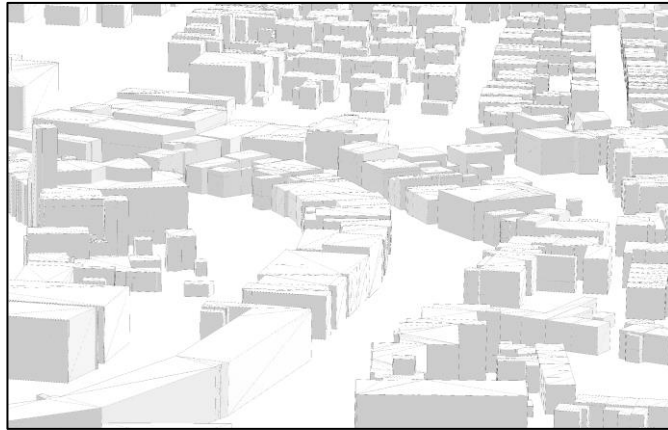
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	107370	Stuttgart		1	48.833	9.2	318								
2	Date (MM/DD/YYYY)	Time (HH:MM)	ETR (W/m^2)	ETRN (W/m^2)	GHI (W/m^2)	GHI source	GHI uncert (%)	DNI (W/m^2)	DNI source	DNI uncert (%)	DHI (W/m^2)	DHI source	DHI uncert (%)	GH illum (lx)	GH illum source
3	01/01/2005	01:00	0	0	0	3	0	0	3	0	0	3	0	0	3
4	01/01/2005	02:00	0	0	0	3	0	0	3	0	0	3	0	0	3
5	01/01/2005	03:00	0	0	0	3	0	0	3	0	0	3	0	0	3
6	01/01/2005	04:00	0	0	0	3	0	0	3	0	0	3	0	0	3
7	01/01/2005	05:00	0	0	0	3	0	0	3	0	0	3	0	0	3
8	01/01/2005	06:00	0	0	0	3	0	0	3	0	0	3	0	0	3
9	01/01/2005	07:00	0	0	0	3	0	0	3	0	0	3	0	0	3
10	01/01/2005	08:00	0	0	0	3	0	0	3	0	0	3	0	0	3
11	01/01/2005	09:00	35	426	4	3	10	0	3	20	4	3	10	546	3
12	01/01/2005	10:00	198	1412	94	3	10	363	3	20	43	3	10	9684	3
13	01/01/2005	11:00	331	1412	174	3	10	475	3	20	63	3	10	18248	3

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	107370	Stuttgart		1	48.833	9.2	318								
2	Date (MM/D	Time (HH:MM	NaN	NaN	GHI (W/m^2)	NaN	NaN	NaN	NaN	NaN	DHI (W/m^2)	NaN	NaN	NaN	NaN
3	01/01/2005	01:00	0	0	0	3	0	0	3	0	0	3	0	0	3
4	01/01/2005	02:00	0	0	0	3	0	0	3	0	0	3	0	0	3
5	01/01/2005	03:00	0	0	0	3	0	0	3	0	0	3	0	0	3
6	01/01/2005	04:00	0	0	0	3	0	0	3	0	0	3	0	0	3
7	01/01/2005	05:00	0	0	0	3	0	0	3	0	0	3	0	0	3
8	01/01/2005	06:00	0	0	0	3	0	0	3	0	0	3	0	0	3
9	01/01/2005	07:00	0	0	0	3	0	0	3	0	0	3	0	0	3
10	01/01/2005	08:00	0	0	0	3	0	0	3	0	0	3	0	0	3
11	01/01/2005	09:00	35	426	4	3	10	0	3	20	4	3	10	546	3
12	01/01/2005	10:00	198	1412	94	3	10	363	3	20	43	3	10	9684	3

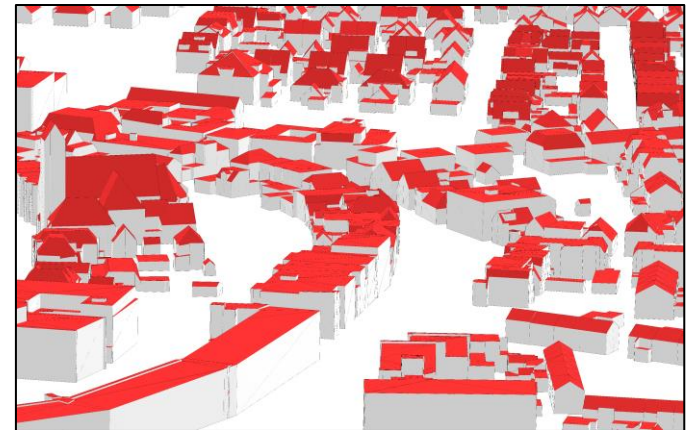
Input data: 3D city model

Building data

- A valid (XML-encoded) CityGML file
 - Geometries are validated upon import by CityDoctor (Coors et al., 2020)
 - LoD1
 - LoD2
 - Minimum set of attributes is discussed in upcoming slides



LoD1



LoD2

Input data: 3D city model

Building data

To perform energy demand simulations, buildings must have at least information for:

- Year of construction
- Usage

Please note: building usage must be written in attribute "function" (so, neither building "class", nor building "usage"!) ←

```
23 | <core:cityObjectMember>
24 | | <bldg:Building gml:id="NL.IMBAG.Pand.174210000004313">
25 | | | <gen:stringAttribute name="ground_height">
26 | | | | <gen:value>11.495</gen:value>
27 | | | </gen:stringAttribute>
28 | | | <gen:stringAttribute name="lod2_area">
29 | | | | <gen:value>18.7</gen:value>
30 | | | </gen:stringAttribute>
31 | | | <gen:stringAttribute name="lod2_volume">
32 | | | | <gen:value>62.39</gen:value>
33 | | | </gen:stringAttribute>
34 | | | <bldg:class>woonfunctie</bldg:class>
35 | | | <bldg:yearOfConstruction>1948</bldg:yearOfConstruction>
36 | | | <bldg:roofType>slanted</bldg:roofType>
```

← This will not work!

Input data: 3D city model

Currently, SimStadt supports only the German encoding definitions – ALKIS (AdV, 2018)

For non-German buildings, **two ways** to set up the function values properly:

- 1) Edit/Update the function attribute(s) in the CityGML file and substitute it with the proper numeric value as per ALKIS codelist
- 2) Write an XML-based mapping file that SimStadt uses to convert "on the fly" the values in the CityGML document to the corresponding ALKIS values
 - This mapping file must be named **AlkisCodes.xml**

Input data: 3D city model

Building data

- 1) Edit/Update the function attribute(s) in the CityGML file and substitute it with the proper numeric value as per ALKIS codelist

```

23     <core:cityObjectMember>
24         <bldg:Building gml:id="NL.IMBAG.Pand.1742100000004313">
25             <gen:stringAttribute name="ground_height">
26                 <gen:value>11.495</gen:value>
27             </gen:stringAttribute>
28             <gen:stringAttribute name="lod2_area">
29                 <gen:value>18.7</gen:value>
30             </gen:stringAttribute>
31             <gen:stringAttribute name="lod2_volume">
32                 <gen:value>62.39</gen:value>
33             </gen:stringAttribute>
34             <bldg:class>woonfunctie</bldg:class>
35             <bldg:function>1010</bldg:function>
36             <bldg:yearOfConstruction>1948</bldg:yearOfConstruction>
37             <bldg:roofType>slanted</bldg:roofType>

```



1) Function as ALKIS code

Input data: 3D city model

Building data

2) Write an XML-based mapping file that SimStadt uses to convert "on the fly" the values in the CityGML document to the corresponding ALKIS values

```

25     <bldg:Building gml:id="NL.IMBAG.Pand.1742100000005418">
26         <gml:name>Building_1200</gml:name>
27         <core:relativeToTerrain>entirelyAboveTerrain</core:relativeToTerrain>
28         <gen:stringAttribute name="pand_id">
29             <gen:value>1742100000005418</gen:value>
30         </gen:stringAttribute>
31         <bldg:class>Non-residential (single-function)</bldg:class>
32         <bldg:function>overige gebruiksfunctie</bldg:function>
33         <bldg:yearOfConstruction>1979</bldg:yearOfConstruction>
34         <bldg:roofType>single horizontal</bldg:roofType>
35         <bldg:lod0FootPrint>

```

```

1     <?xml version="1.0" encoding="ISO-8859-1"?>
2
3     <codes xmlns:buLib="http://www.simstadt.eu/BuildingUsageLibraries" xmlns="http://www.simstadt.eu/AlkisCodes">
4
5     <codesForBuildingUsageLibraryNamed>Germany_DIN18599</codesForBuildingUsageLibraryNamed>
6
7     <code from="overige gebruiksfunctie" to="9997"/>
8
9 </codes>

```



2) CityGML with AlkisCodes.xml

Input data: libraries

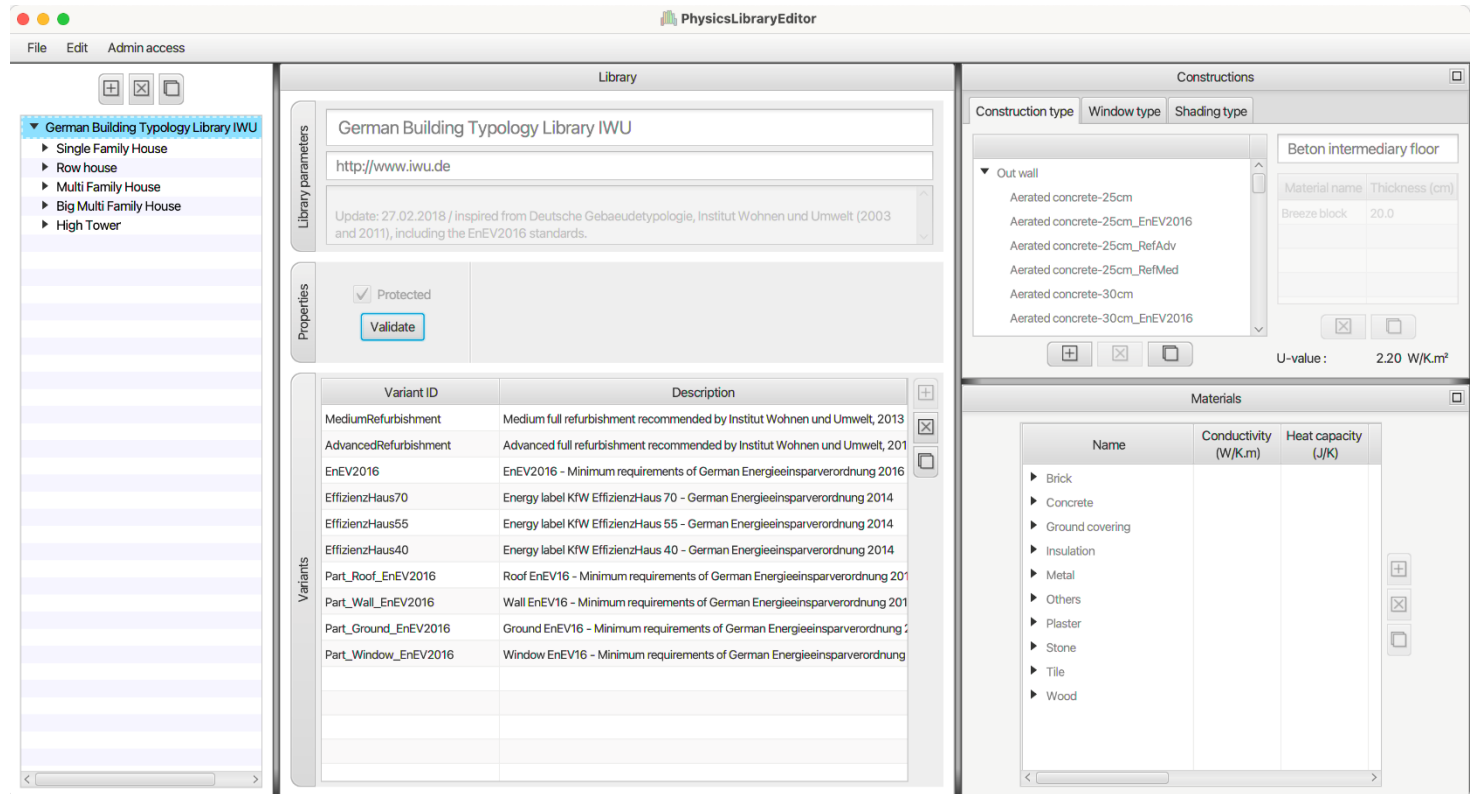
Building physics library

- Building physics parameters are required for energy demand simulations
- The building physics library contains parameters regarding
 - Constructions & Materials
 - U-values of the different building elements
 - G-values of glazing surfaces
 - Average thermal bridges, etc.
- Currently SimStadt only supports building parameters for German and NYC typologies
- (It seems that) it can be customised
 - ...but currently it is not clear how it should be done and whether it works...
 - For the time being, simply use the existing library

Input data: libraries

Building physics library

- Introduction
- Installation
- Fundamentals
- Input**
- Output
- SimStadt Workflows
- References



The screenshot shows the PhysicsLibraryEditor interface with the following sections:

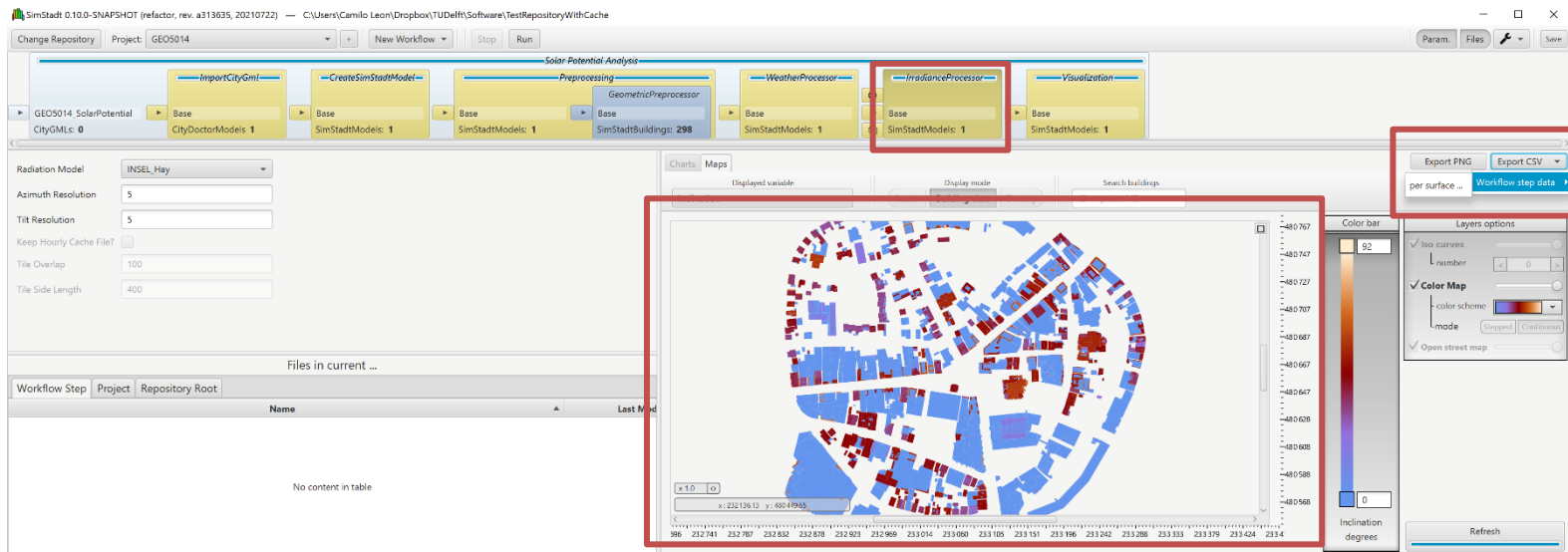
- Library:** German Building Typology Library IWU, URL: <http://www.iwu.de>, Update: 27.02.2018 / inspired from Deutsche Gebäudetypologie, Institut Wohnen und Umwelt (2003 and 2011), including the EnEV2016 standards.
- Properties:** Protected,
- Variants:**

Variant ID	Description
MediumRefurbishment	Medium full refurbishment recommended by Institut Wohnen und Umwelt, 2013
AdvancedRefurbishment	Advanced full refurbishment recommended by Institut Wohnen und Umwelt, 201
EnEV2016	EnEV2016 - Minimum requirements of German Energieeinsparverordnung 2016
EffizienzHaus70	Energy label KfW EffizienzHaus 70 - German Energieeinsparverordnung 2014
EffizienzHaus55	Energy label KfW EffizienzHaus 55 - German Energieeinsparverordnung 2014
EffizienzHaus40	Energy label KfW EffizienzHaus 40 - German Energieeinsparverordnung 2014
Part_Roof_EnEV2016	Roof EnEV16 - Minimum requirements of German Energieeinsparverordnung 201
Part_Wall_EnEV2016	Wall EnEV16 - Minimum requirements of German Energieeinsparverordnung 201
Part_Ground_EnEV2016	Ground EnEV16 - Minimum requirements of German Energieeinsparverordnung 201
Part_Window_EnEV2016	Window EnEV16 - Minimum requirements of German Energieeinsparverordnung 201
- Constructions:**
 - Construction type: Window type
 - Selected: Out wall
 - Material list:

Material name	Thickness (cm)
Breeze block	20.0
 - U-value: 2.20 W/K.m²
- Materials:**
 - Table with columns: Name, Conductivity (W/K.m), Heat capacity (J/K)
 - Material list: Brick, Concrete, Ground covering, Insulation, Metal, Others, Plaster, Stone, Tile, Wood

Data export

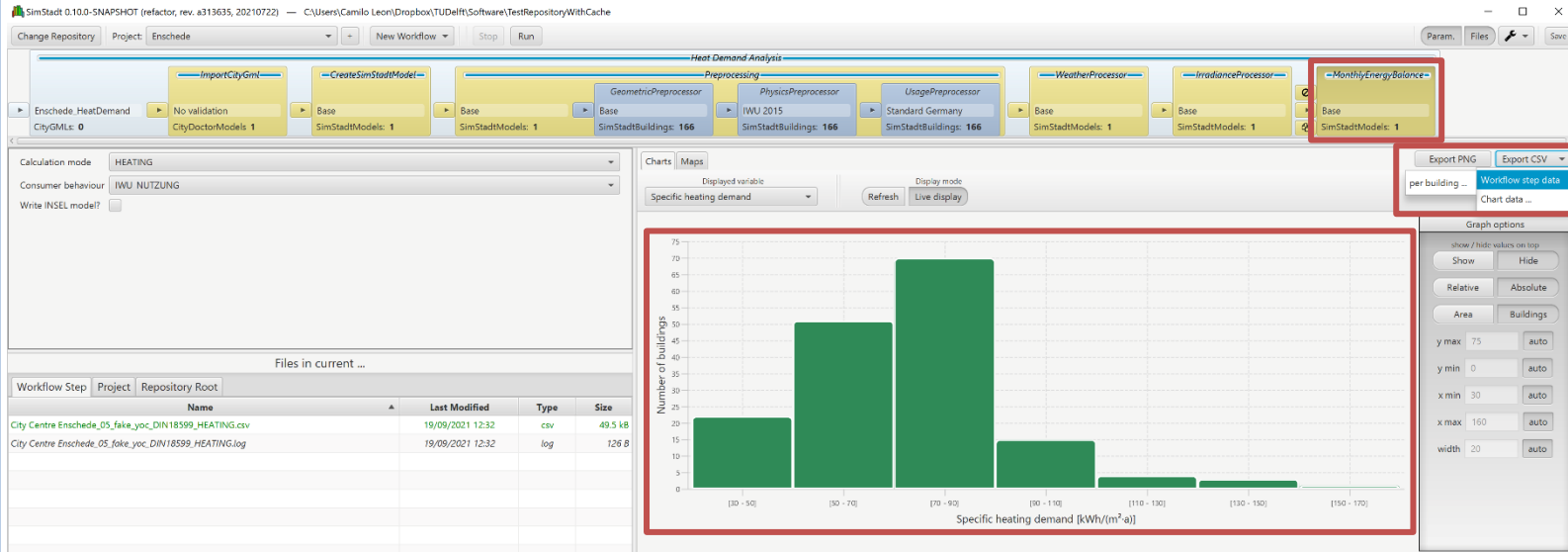
- Using the GUI, results from each processing step can be exported as:
 - PNG images (current visualisation)
 - CSV files



IrradianceProcessor

Data export

- Using the GUI, results from each processing step can be exported as:
 - PNG images (current visualisation)
 - CSV files



The screenshot shows the SimStad GUI interface. The main workflow is titled 'Heat Demand Analysis' and includes steps: ImportCityGrid, CreateSimStadModel, Preprocessing (GeometricPreprocessor, PhysicsPreprocessor, UsagePreprocessor), WeatherProcessor, IrradianceProcessor, and MonthlyEnergyBalance. The 'MonthlyEnergyBalance' step is highlighted with a red box. Below the workflow, there are settings for Calculation mode (HEATING), Consumer behaviour (IWU NUTZUNG), and Write INSEL model? (unchecked). A 'Files in current ...' table is visible at the bottom left, showing files generated during the simulation.

Workflow Step	Project	Repository Root	Name	Last Modified	Type	Size
	City Centre Enschede	05_fake_yoc_DIN18599	HEATING.csv	19/09/2021 12:32	csv	49.5 kB
	City Centre Enschede	05_fake_yoc_DIN18599	HEATING.log	19/09/2021 12:32	log	126 B

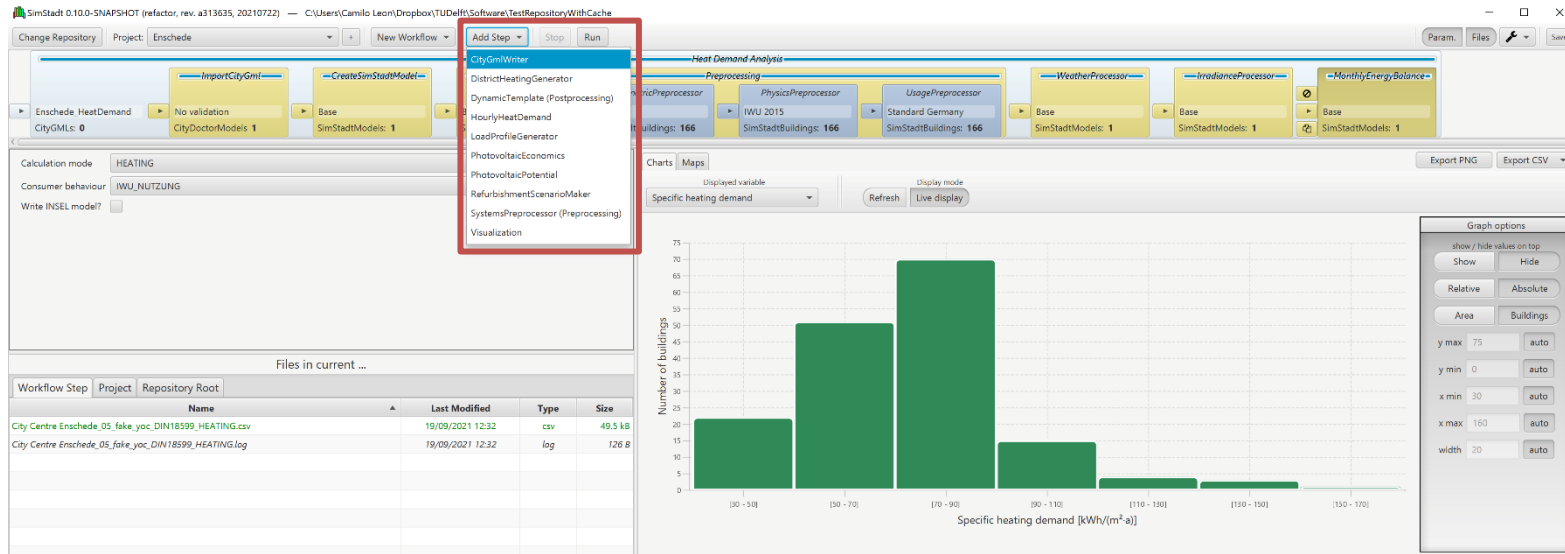
The bar chart displays the distribution of specific heating demand (kWh/(m²·a)) across buildings. The x-axis represents the specific heating demand ranges, and the y-axis represents the number of buildings. The distribution is as follows:

Specific heating demand [kWh/(m²·a)]	Number of buildings
[30 - 50]	22
[50 - 70]	40
[70 - 90]	68
[90 - 110]	15
[110 - 130]	5
[130 - 150]	3
[150 - 170]	2

MonthlyEnergyBalance

Data export: CityGML

- There are two possibilities to export results in CityGML
 - a) Adding a new *CityGMLWriter* step to the workflow



The screenshot shows the SimStadt 0.10.0-SNAPSHOT interface. The workflow editor at the top displays a sequence of steps: ImportCityGml, CreateSimStadtModel, No validation, CityDoctorModels, Base, SimStadtModels, PhysicsPreprocessor, UsagePreprocessor, WeatherProcessor, IrradianceProcessor, and MonthlyEnergyBalance. A red box highlights the 'Add Step' dropdown menu, with 'CityGMLWriter' selected. Below the workflow editor, the 'Files in current ...' table is visible:

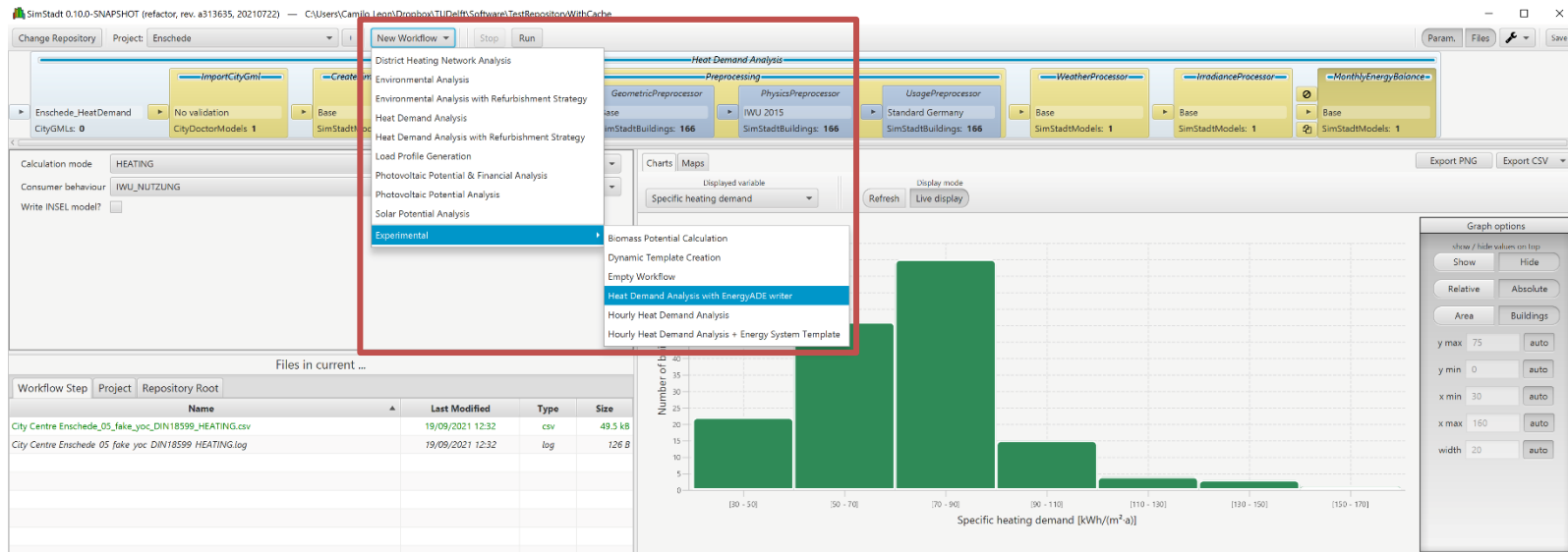
Workflow Step	Project	Repository Root	Name	Last Modified	Type	Size
			City Centre Enschede_05_fake_yoc_DIN18599_HEATING.csv	19/09/2021 12:32	csv	49.5 kB
			City Centre Enschede_05_fake_yoc_DIN18599_HEATING.log	19/09/2021 12:32	log	126 B

The bottom right of the interface shows a bar chart titled 'Specific heating demand [kWh/(m²·a)]'. The x-axis represents specific heating demand ranges, and the y-axis represents the number of buildings. The chart shows a distribution with a peak in the 70-90 kWh/(m²·a) range.

Step CityGMLWriter

Data export: CityGML (+ Energy ADE)

- There are two possibilities to export results in CityGML
 - b) Using the specific workflow "Heat Demand Analysis with the Energy ADE writer"



The screenshot shows the SimStadt software interface. A red box highlights the 'New Workflow' menu, where 'Heat Demand Analysis with the Energy ADE writer' is selected. The main workspace displays a workflow diagram with steps like 'ImportCityGml', 'CreateCityGML', 'GeometricPreprocessor', 'PhysicsPreprocessor', 'UsagePreprocessor', 'WeatherProcessor', 'IrradianceProcessor', and 'MonthlyEnergyBalance'. A bar chart on the right shows the distribution of specific heating demand in kWh/m²·a.

Workflow Step	Project	Repository Root	Name	Last Modified	Type	Size
			City Centre Enschede_05_fake_yoc_DIN18599_HEATING.csv	19/09/2021 12:32	csv	49.5 kB
			City Centre Enschede_05_fake_yoc_DIN18599_HEATING.log	19/09/2021 12:32	log	126 B

Heat Demand Analysis with the Energy ADE writer

CityGML + Energy ADE

Please note: as of **autumn 2024**, the exported CityGML + Energy ADE file

- contains only data according to the Energy ADE KIT profile
 - <https://www.citygmlwiki.org/images/4/41/KIT-UML-Diagramme-Profil.pdf>
- contains some errors and is therefore not valid. For example:
 - The tag `energy:timeInterval` contains the value "month". It is not a valid Time Unit Type
 - A workaround is to express it as 0.083 of a year
 - FloorArea and EnergyDemand elements are missing the corresponding property tag

If you want to go this way, it is therefore recommended to adjust/correct the file (e.g. in FME). Only once the file is valid, it can be imported in the 3DCityDB with the Energy ADE plugin.

More details can be found in:

León-Sánchez, C., Giannelli, D., Agugiaro, G., Stoter, J., 2021,

Testing the new 3D BAG dataset for energy demand estimation of residential buildings.

ISPRS Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLVI-4/W1-2021, pp. 69–76

<https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLVI-4-W1-2021/69/2021/> (open access)

CityGML + Energy ADE

- Introduction
- Installation
- Fundamentals
- Input
- Output
- SimStadt Workflows
- References

```

54615 </bldg:WallSurface>
54616 </bldg:boundedBy>
54617 <energy:FloorArea>
54618 <energy:type>energyReferenceArea</energy:type>
54619 <energy:value uom="m2">4.200314056665326</energy:value>
54620 </energy:FloorArea>
54621 <energy:EnergyDemand>
54622 <energy:energyAmount>
54623 <energy:RegularTimeSeries>
54624 <energy:variableProperties>
54625 <energy:TimeValuesProperties>
54626 <energy:acquisitionMethod>simulation</energy:acquisitionMethod>
54627 <energy:source>SimStadt 0.10.0-SNAPSHOT (refactor, rev. a313635, 20210722)</energy:source>
54628 </energy:TimeValuesProperties>
54629 </energy:variableProperties>
54630 <energy:timeInterval unit="month">1.0</energy:timeInterval>
54631 <energy:values uom="kWh">5.668709945894079E-11 7.408296198718745E-12 4.718447854656915E-15 0.0 0.0 0.0 0.0 0.0 0.0
8.859579736508749E-14 2.8244073746463982E-11 7.866063356232189E-11</energy:values>
54632 </energy:RegularTimeSeries>
54633 </energy:energyAmount>
54634 <energy:endUse>spaceHeating</energy:endUse>
54635 </energy:EnergyDemand>
54636 <energy:EnergyDemand>
54637 <energy:energyAmount>
54638 <energy:RegularTimeSeries>
54639 <energy:variableProperties>
54640 <energy:TimeValuesProperties>
54641 <energy:acquisitionMethod>simulation</energy:acquisitionMethod>
54642 <energy:source>SimStadt 0.10.0-SNAPSHOT (refactor, rev. a313635, 20210722)</energy:source>
54643 </energy:TimeValuesProperties>
54644 </energy:variableProperties>
54645 <energy:timeInterval unit="year">1.0</energy:timeInterval>
54646 <energy:values uom="kWh">0.0</energy:values>
54647 </energy:RegularTimeSeries>
54648 </energy:energyAmount>
54649 <energy:endUse>domesticHotWater</energy:endUse>
54650 </energy:EnergyDemand>
54651 </bldg:Building>
54652 </core:cityObjectMember>

```

It should be inside tag
energy:floorArea

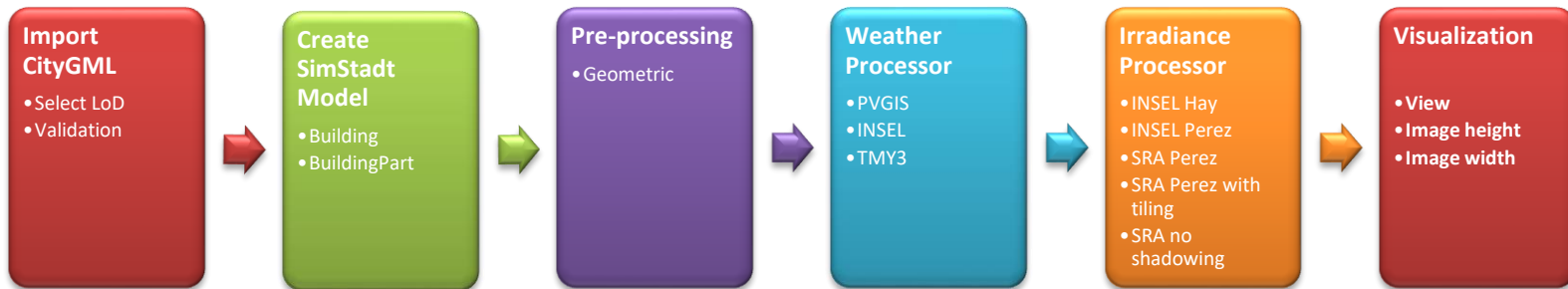
It should be inside tag
energy:demands

Incorrect time interval

Example of some errors in the
CityGML + Energy ADE file
from SimStadt

Solar analysis

These are the typical steps that compose the "Solar Potential analysis" workflow in SimStadt



Sky Model
John Hay
Richard Perez

Irradiance processor

When performing solar analysis, some specific simulation parameters must be set. Here some of the most relevant ones.

Of particular relevance is whether the effect of shadow-casting objects must be considered or not. SimStadt offers the following radiation models.

Radiation models

- **INSEL_Hay**: computationally fast, without shadows
- **INSEL_Perez**: computationally fast, more detailed and possibly more accurate than Hay, without shadows
- **Perez Simplified Radiosity Algorithm**: computationally slow, should not be used for larger models, considering shadows
- **Perez Simplified Radiosity Algorithm on tiles**: optimised for larger models (split in smaller tiles), considering shadows
- **Perez Simplified Radiosity Algorithm on a geodesic dome**: computationally fast, can be used for comparisons with other SRA calculations, without shadows

Irradiance processor

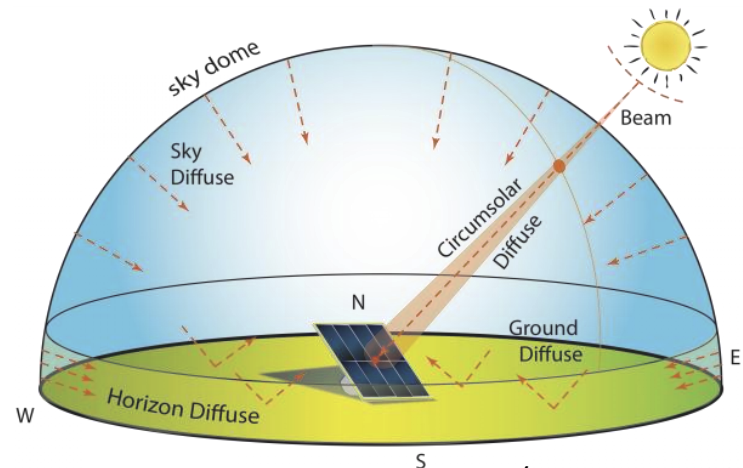
When performing solar analysis, some specific simulation parameters must be set. Here some of the most relevant ones.

For INSEL:

The sky dome is discretised, and irradiance is computed considering a "grid" defined by azimuth and tilt (inclination) resolution. Both values indicate the "size" in decimal degrees of each grid cell.

- Azimuth resolution: **int, unit [°]**
 - e.g. 5°
- Tilt resolution: **int, unit [°]**
 - e.g. 5°

Please note: the smaller the values of the angles, the longer the time needed by the simulation (obviously!)



(Brownson, 2020)

Irradiance processor

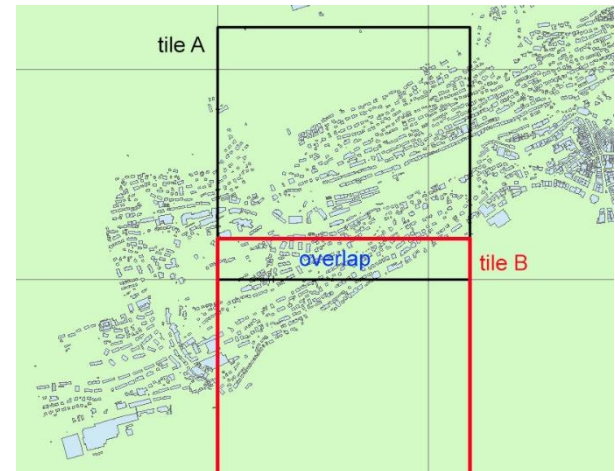
When performing solar analysis, some specific simulation parameters must be set. Here some of the most relevant ones.

For SRA_Perez_with_tiling

The complete 3D model is split in smaller tiles before shadows are calculated. In order to consider shadows for the buildings located on the edge of the tiles, tiles must overlap

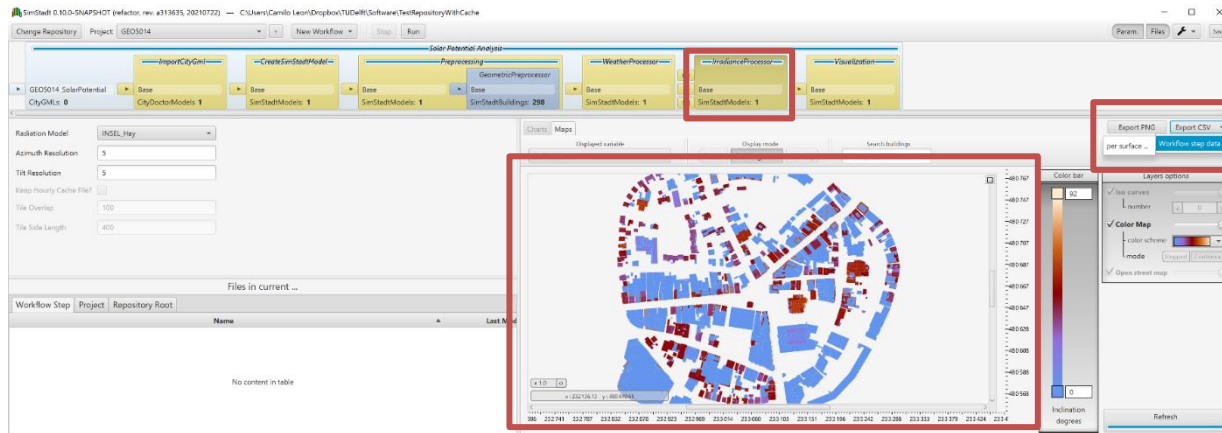
- Tile side: **int, unit [m]**
 - e.g. 400 m
- Tile overlap: **int, unit [m]**
 - e.g. 100 m

Please note: if you use tiling, you **MUST** provide an overlap values > 0 to assure that no buildings are missed



Irradiance processor

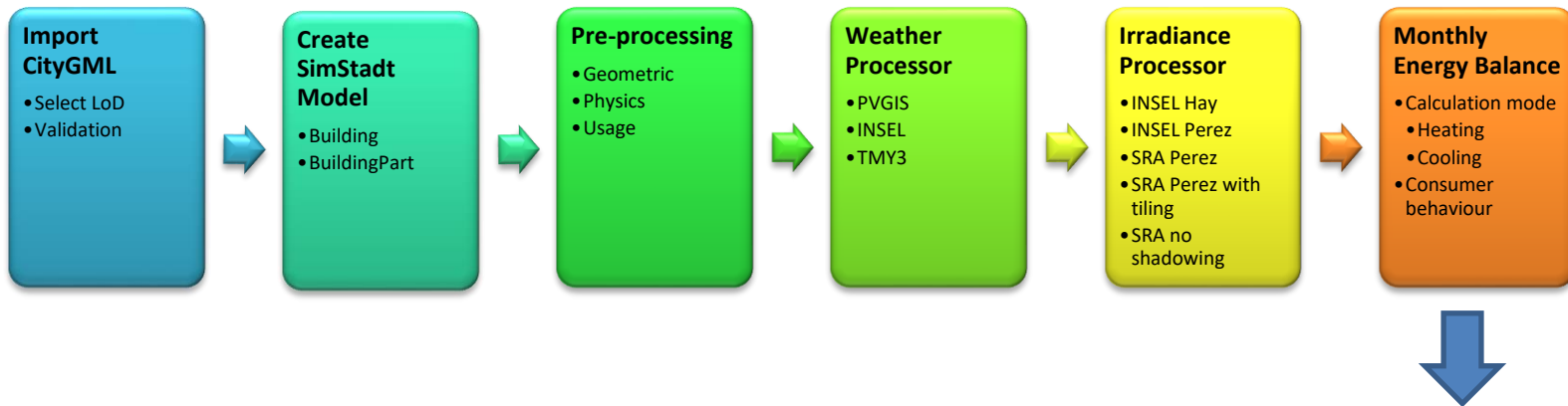
- Results are only available as PNG and CSV
- The exported csv file looks like:



- 1 Surface ID, Building ID, Type, Azimuth [°], Inclination [°], Area [m2], Irradiance [W/m2]
- 2 WallSurface_UUID_42d30a68-22bf-4fc2-a042-e089b1843de1, NL.IMBAG.Pand.174210000005721, WALL, 263.00, 90.00, 11.95, 79.10
- 3 RoofSurface_UUID_58c1509e-6b37-4b8c-8890-edbcaaca02cb, NL.IMBAG.Pand.174210000005721, ROOF, 180.00, 0.00, 44.15, 118.73
- 4 WallSurface_UUID_34860986-b28e-4005-8a4b-b3d5af196933, NL.IMBAG.Pand.174210000005721, WALL, 173.00, 90.00, 28.21, 92.12
- 5 WallSurface_UUID_bf60ac11-be10-437f-b06c-00585dd9a392, NL.IMBAG.Pand.174210000005721, WALL, 352.00, 90.00, 0.00, 0.00
- 6 WallSurface_UUID_0ad442b8-c7c6-4b4f-bc3a-20867d78797a, NL.IMBAG.Pand.174210000005721, WALL, 83.00, 90.00, 12.29, 75.87
- 7 GroundSurface_UUID_1b3ec8f8-f907-44a5-a05e-ab95522b3ed5, NL.IMBAG.Pand.174210000005721, GROUND, 180.00, 180.00, 44.15, 0.00
- 8 RoofSurface_UUID_54778d6f-0704-49a5-8433-e8cbe77db2a3, NL.IMBAG.Pand.174210000005720, ROOF, 180.00, 0.00, 47.61, 118.73
- 9 WallSurface_UUID_2b4f15f8-d16f-4053-b5c2-cb11ffe3669d, NL.IMBAG.Pand.174210000005720, WALL, 353.00, 90.00, 28.75, 52.40
- 10 WallSurface_UUID_e4d0cd42-6930-4689-b7f0-458cc311b9ec, NL.IMBAG.Pand.174210000005720, WALL, 83.00, 90.00, 13.11, 75.87

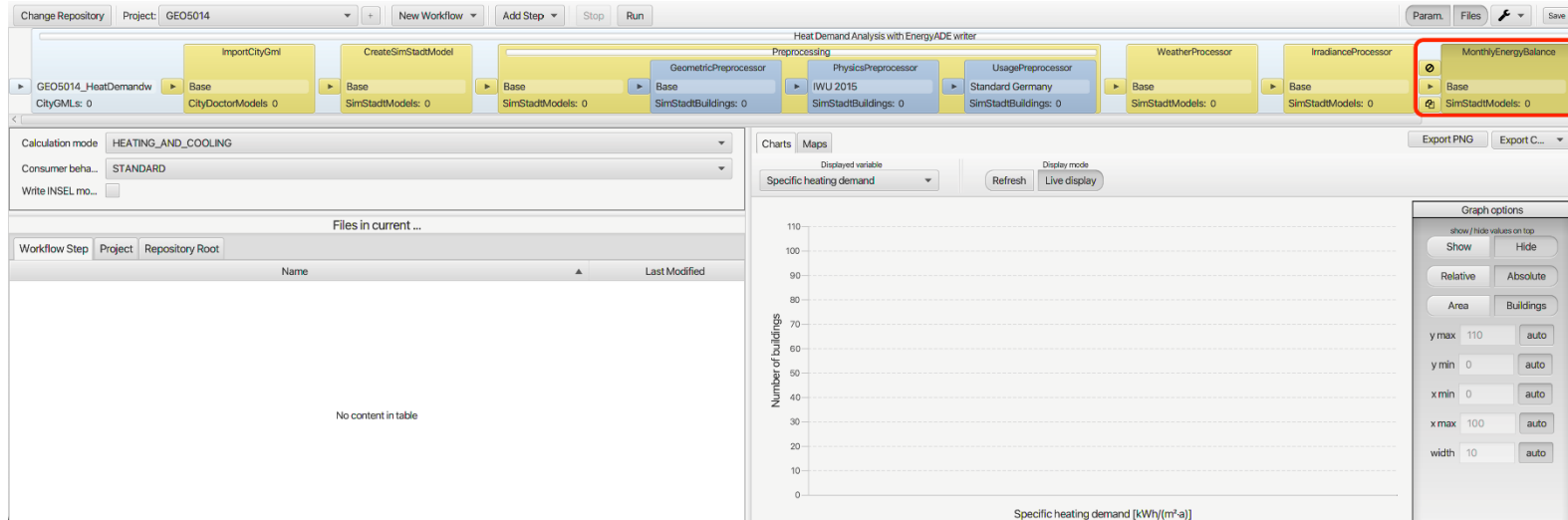
Energy demand analysis

These are the typical steps that compose the "Heat Demand Analysis" workflow in SimStadt



Please note: Heating analysis includes domestic hot water (DHW) production

Energy demand analysis



Overview of the "Heat Demand Analysis" workflow

Energy demand analysis

- Results can be exported as
 - CSV file
 - CityGML + Energy ADE (see previous slides)

```

1  ## SimStadt DIN18599 Monthly Energy Balance
2  #
3  # Weather data source;meteonorm tmy3 file 2000-2009
4  # Location;Twenthe Enschede AP
5  # Coordinate reference system;EPSG:28992
6  # Latitude;52.27;°
7  # Longitude;6.89;°
8  # Radiation model;meteonorm_Hay
9  # Calculation Mode;HEATING_AND_COOLING
10 # Consumer behaviour;IWU_NUTZUNG
11 #
12 # Building physics library;German Building Typology Library IWU (http://www.iwu.de)
13 # Building usage library;Building Usage Library (based on UsageLibrary.xlsx, sheet: Germany)
14 # Has custom building usage codes?;true
15 #
16 # SimStadt version;0.10.0-SNAPSHOT (refactor, rev. a313635, 20210722)
17 # Created at;2021/09/19 21:05:14
18 # Cache version;7
19 #
20 GMLId;ParentGMLId;Latitude;Longitude;X-coordinate;Y-coordinate;LOD;Year of construction;Year of refurbishment;Refurbishment Variant;ALKIS
21 [-]; [-]; [°]; [°]; [?]; [?]; LOD; [YYYY]; [YYYY]; [-]; [-]; [-]; [m²]; [-]; [m²]; [-]; [m²]; [m²]; [m²]; [m²]; [m²]; [m²]; [m²]; [m²]; [m²]; [m²]; [0/1]; [m³]; [m]; [m]; [-]; [m]; [-]
22 NL.IMBAG.Pand.1742100000005721;;52.30851;6.52004;232256.01;480665.09;LOD2;1996;;Original;1630;non-heated;35.8;none;0.0;RH;44.2;47.8;0.0;5
23 NL.IMBAG.Pand.1742100000005720;;52.30855;6.52003;232255.46;480669.52;LOD2;1996;;Original;1630;non-heated;38.8;none;0.0;RH;47.6;50.3;0.0;5
24 NL.IMBAG.Pand.1742100000004755;;52.30660;6.52066;232301.40;480452.80;LOD2;1948;;Original;1000;residential;127.4;none;0.0;EFH;138.3;185.4;
25 NL.IMBAG.Pand.1742100000005723;;52.30851;6.52046;232284.49;480665.48;LOD2;1933;;Original;1000;residential;131.8;none;0.0;RH;104.4;144.5;0
26 NL.IMBAG.Pand.1742100000005722;;52.30843;6.52041;232281.42;480656.06;LOD2;1933;;Original;1000;residential;20.9;none;0.0;RH;35.2;58.2;0.0;
27 NL.IMBAG.Pand.1742100000004754;;52.30677;6.52074;232306.67;480471.86;LOD2;1948;;Original;1000;residential;242.3;none;0.0;EFH;129.4;261.3;

```

Example of heat demand analysis results exported as CSV file

- Weiler, V., Stave, J., & Eicker, U. (2019). **Renewable energy generation scenarios using 3D urban modeling tools—methodology for heat pump and co-generation systems with case study application.** *Energies*, 12(3). <https://doi.org/10.3390/en12030403>
- Coors, V., Betz, M., & Duminil, E. (2020). **A Concept of Quality Management of 3D City Models Supporting Application-Specific Requirements.** *PFG - Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 88(1), 3–14. <https://doi.org/10.1007/s41064-020-00094-0>
- AdV. (2018). **Dokumentation zur Modellierung der Geoinformationen des amtlichen Vermessungswesens (GeoInfoDok): Vol. 7.1 rc.1.** <http://www.geodatenzentrum.de/docpdf/ATKIS-OK Basis-DLM 6 0.pdf>
- EnergyPlus. (2021). **Weather Data.** <https://energyplus.net/weather>
- NREL. (2021). **NSRDB: National Solar Radiation Database.** <https://nsrdb.nrel.gov/>
- EU Science Hub. (2021). **Photovoltaic Geographical Information System (PVGIS).** <https://ec.europa.eu/jrc/en/pvgis>
- Brownson, J. (2020). **Solar Resource Assessment and Economics.** <https://www.e-education.psu.edu/eme810/node/679>



Camilo León-Sánchez

c.a.leonsanchez@tudelft.nl

3D Geoinformation Group

TU Delft

The Netherlands

<https://3d.bk.tudelft.nl/cleon>



Dr. Giorgio Agugiaro

g.agugiaro@tudelft.nl

3D Geoinformation Group

TU Delft

The Netherlands

<https://3d.bk.tudelft.nl/gagugiaro>

Acknowledgements

Eric Duminil, Volker Coors (HFT Stuttgart)