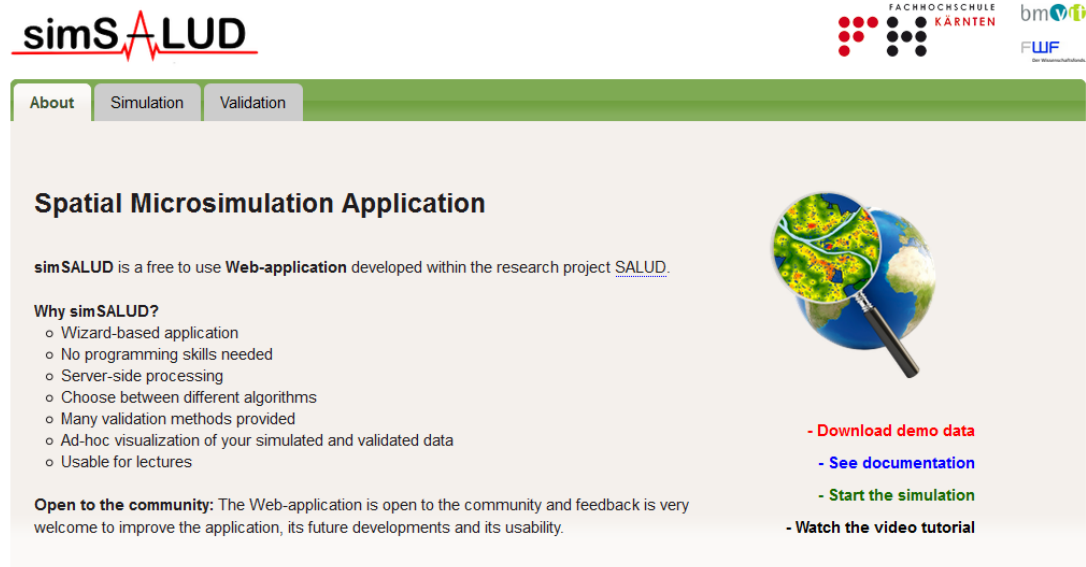


simSALUD (v. 1.3) setup for Windows within Eclipse IDE



Including Modules:

- Simulation
 - o Combinatorial Optimisation
 - o Iterative Proportional Fitting
- Validation
 - o Total Absolute Error
 - o Total Absolute Error - Percent of Total Regions
 - o Standardized Absolute Error
 - o Percentage Error
 - o Independent Samples T-Test
 - o Correlation Coefficient
 - o Simple Regression
- Visualization
 - o Visualize Simulation Results
 - o Visualize Validation Results
 - o Upload and Visualize Simulated Data

simSALUD setup with visualization

Authors: B. Kosar & M. Tomintz

May 2015

simSALUD (v. 1.3) setup for Windows within Eclipse IDE

A SALUD White Paper

Contents	Page
About This Tutorial	Error! Bookmark not defined.
What You Will Need	Error! Bookmark not defined.
What You Will Do	Error! Bookmark not defined.
Download Programs and Files	Error! Bookmark not defined.
Install PostgreSQL 9.2.10.....	Error! Bookmark not defined.
Configure PostgreSQL.....	Error! Bookmark not defined.
Install the Java Development Kit and Runtime Environment	Error! Bookmark not defined.
Install Apache Tomcat 7	Error! Bookmark not defined.
Install eclipse IDE.....	Error! Bookmark not defined.
Configure eclipse IDE – Set up server.....	Error! Bookmark not defined.
Configure eclipse IDE – Choose web browser	Error! Bookmark not defined.
Configure eclipse IDE – Import simsalud.war file.....	Error! Bookmark not defined.
Test the simulation.....	Error! Bookmark not defined.
Install ArcGIS for Server 10.3.....	5
Connect ArcGIS Server in ArcMap for publishing spatial data	5
Set up a geodatabase in PostgreSQL on Windows.....	5
Registering the simsalud PostgreSQL database with the server	Error! Bookmark not defined.
Registering the spatial ArcSDE database with the server	Error! Bookmark not defined.
Adding a new shape file to the simSALUD web application (Publishing a service).....	6
Joining both databases.....	7
Add the required map parameters in eclipse.....	8
Use simsalud on localhost	Error! Bookmark not defined.
Error handling.....	10
Appendix 1: Software and Download link required in this tutorial.....	Error! Bookmark not defined.
Appendix 2: Database Structure	Error! Bookmark not defined.
Appendix 3: Java packages	Error! Bookmark not defined.

simSALUD (v. 1.3) setup for Windows within Eclipse IDE

simSALUD is a free to use Web-application developed within the research project SALUD (SpatiAL microsimULation for health Decision support. SALUD is funded by the Federal Ministry for Transport, Innovation and Technology (bmvit) and the Austrian Science Fund (FWF) at the Carinthia University of Applied Sciences, Department of Geoinformation and Environmental Technology in Austria. For more information visit www.simsalud.org.

About This Tutorial

This document guides through setting up the simSALUD web-application on your local machine.

This tutorial assumes you can do the following:

- Navigate your computer's file system with Windows Explorer
- Installing software
- Open a web browser and enter a URL

What You Will Need (Requirements)

Windows 7 (at least)
PostgreSQL
Java Development Kit (JDK) and Runtime Environment (Eclipse IDE)
Apache Tomcat Servlet
Administrator privileges
Internet connection (to download the required software)

All required software packages are available within this guide for installation (see folder "software"). Download the programs and files listed in table 1 by searching online, using the enclosed software for the specific versions listed or using the links provided in appendix 1.

What You Will Do

1. Download the required software
2. Install PostgreSQL
3. Install the Java Development Kit and Runtime Environment
4. Install Eclipse IDE
5. Install Apache Tomcat servlet
6. Set up the user and schema for the simSALUD application

Table 1: Software required for installing simSALUD

Component	Function	Version	File / URL
PostgreSQL	RDBMS	9.2.x	http://www.postgresql.org/
Java Development Kit with Runtime Environment	Java controller	8ux	http://oracle.com/java
Apache Tomcat	Web servlet	7.0.x	http://tomcat.apache.org/
Eclipse	IDE	Luna	https://eclipse.org/

Install PostgreSQL 9.2.x

The PostgreSQL RDBMS will store all uploaded, simulated and validated data of the user when using simSALUD.

1. Go to PostgreSQL website – Download – Windows – Download - download the Installer version 9.2.x for either the 32bit (Win x86-32) or the 64bit (Win x86-64) version (check your system requirements if you are not sure).
2. Run the Windows installer for PostgreSQL version 9.2.x. Accept all defaults.
3. When asked to create a password for the postgres user, enter *postgres*.
4. Verify that the PostgreSQL port number is 5432.
5. In case “Stack Builder” pops up to install, then press “cancel”.
6. The PostgreSQL user interface “pgAdmin III” was installed a well.

Configure PostgreSQL

1. Open the software “pgAdmin III” – Double click at “PostgreSQL 9.2 (localhost:5432)
2. Enter the password (default: *postgres*)
3. Right click on “Login Roles” – New Login Role... – Role name: *salud*.
4. Move on to the tab “Definition” and type in the password: *salud* – OK
5. Right click on “Databases” – New Database... - Name: *simсалуд* – Owner: *salud* - OK

Install Java Development Kit and Runtime Environment

The JDK includes a private java virtual machine and a few other resources for using the integrated development environment (IDE) Eclipse for developing Java Application.

1. Go to the Oracle Java website – Downloads - Java for Developers – (Java SE 8u45) JDK Download – Accept License Agreement - download the Installer version 8u45 for either the 32bit (jdk-8u45-windows-i586.exe) or the 64bit (jdk-8u45-windows-x64.exe) version.
2. Run the Installer and accept all the defaults.

Install Apache Tomcat 7

Apache Tomcat 7 is web servlet software that manages web applications. It is used to deploy the simSALUD web application.

1. Go to the Apache Tomcat website – Download - Tomcat 7.0 – Binary Distributions – Core: - 32-bit/64-bit Windows Service Installer
2. Verify that the Tomcat port number is 8080 (HTTP)
3. Use “tomcat” as administrator user name and password.
4. Run the installer and accept all the defaults.
5. Uncheck “Run Apache Tomcat” and “Show Readme”

[Skip the next steps and continue with “Error! Reference source not found.” if you don’t want to develop anything]

Install Eclipse IDE

Eclipse is an integrated development environment (IDE) and is used to develop applications

1. Go to the Eclipse website – Download – Download the “Eclipse IDE for Java EE Developers” for either the 32bit or 64bit version.
2. Extract the contents of the *.zip file to your directory (e.g.: C://Programs).

Configure Eclipse IDE – Setup server

Note: If Tomcat 7 is already configured in Eclipse IDE, then skip these steps and move to “Configure Eclipse IDE – Choose web browser”

1. Navigate to your directory and open the Eclipse IDE as Administrator [Right click eclipse.exe – Properties – Compatibility – Check “Execute Program as administrator”] – Use the default settings for selecting a workspace or select a new one – OK
2. After Eclipse IDE opens, right click in the “Project Explorer” window – New – Other
3. Search for “Server” – Select Server – click “Next” – Open the folder “Apache” and select “Tomcat v7.0 Server” – click “Next”
4. Click “Browse...” and navigate to the installation directory of Apache Software and select the folder “Tomcat 7.0”. (e.g.: C:\Program Files\Apache Software Foundation\Tomcat 7.0) – Next – Finish.

Configure Eclipse IDE – Choose web browser

In Eclipse click “Window” – “Web Browser” – Select one external web browser (e.g.: recommended: Firefox, Chrome)

Configure Eclipse IDE – Import simsalud.war file

1. Right click in the “Project Explorer” window – Import – WAR file – Select the “simsalud.war” file and click “Open”
2. Be sure that “Apache Tomcat v7.0” is selected as “Target runtime” and click “Finish”
3. In the Project Explorer expand the following folders “Java Resources” – “src” then expand the package “at.fh.karnten.database” and open: “PostgreSQLAccess.java”
4. Be sure that these parameters are according to the settings in the chapter: “Configure PostgreSQL”.

Test the simulation

Note: In case another server instance is running, a message can occur. If this happens, the server needs to be stopped.

1. In the Project Explorer window, right click on the project “simsalud” – Run As – Run on Server – Check “Always use this server when running this project” – Finish (Grant possible access messages)

The browser should open simSALUD in a new tab and is ready for your simulation.

Hint: In Eclipse (developing the application) it is not possible to download simulated files via the simulation page. Simulated and validated fields are stored here: “C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\simsalud\data\user_...”. For using the application the following steps are required:

Install ArcGIS for Server 10.3 (normally this is done by your IT administrator)

1. Obtain an authorization file for the machine on which you're installing ArcGIS for Server (ask your IT-administrator)
2. Browse to the downloaded folder containing the ArcGIS for Server setup and execute the "ESRI.exe" application
3. Follow the ArcGIS for Server wizard
4. After the installation completes, the Software Authorization Wizard is launched.
5. Check the authorization information
6. Check the authorized software extensions (have to be registered on the web before)
7. After completing authorization, ArcGIS Server Manager should already be open. You can now create a new site. Typically, this is located at <http://<server name>:6080/arcgis/rest/services>.
8. Create a new ArcGIS for Server administrator account
9. Choose the directory in which ArcGIS for Server will store all relevant site information
10. After completing authorization, you can now log in to Manager (You can now check out the preinstalled "SampleWorldCities" geodatabase)

Connect ArcGIS Server in ArcMap for publishing spatial data

This is used to publish all spatial data to the ArcGIS for Server via ArcMap. This can be also used to publish, edit, add, remove, start, and stop services.

1. In the Catalog tree, expand the GIS Servers node.
2. Double-click "Add ArcGIS Server".
3. Choose "Publish GIS services" and click "Next".
4. Enter the Server URL (e.g.: <http://simsalud.cti.ac.at:6080/arcgis>)
5. Enter your User Name (e.g.: *demo*) and Password (e.g.: *demo*). Optionally, you can choose to save your user name and password information so that you don't have to enter it each time you connect.
6. Click "Finish". Your connection appears in the GIS Servers node in the Catalog tree.
7. Optionally, rename your connection. To do so, right-click the connection in the Catalog tree and choose *Rename*.

Set up a geodatabase in PostgreSQL (normally this is done by your IT administrator)

Before performing the "Create Enterprise Geodatabase" geoprocessing tool in ArcMap, PostgreSQL must be installed and configured.

1. Configure the database cluster to accept connections.
2. Start pgAdmin III, connect to the PostgreSQL instance as the postgres super user, connect to the database, click Tools, point to Server Configuration, and then click `pg_hba.conf`.
3. Add the IP addresses of connecting client machines and the client authentication methods used for connections.
4. Restart the PostgreSQL service.
5. Transfer the client files (PostgreSQL/bin) to the bin directory of your ArcGIS client installation.
 - Files are `libeay32.dll`, `libiconv-2.dll`, `libintl.dll`, `libpq.dll`, and `ssleay32.dll`.
6. Place the ST_Geometry libraries in the PostgreSQL lib directory. The `st_geometry.dll` file for use with PostgreSQL 9.2 can be found

in the DatabaseSupport directory of your ArcGIS client installation directory (C:\Program Files (x86)\ArcGIS\Desktop<release#>\DatabaseSupport\PostgreSQL\Windows64). Move the st_geometry.dll file from the ArcGIS client to the PostgreSQL "lib" directory on your PostgreSQL server.

7. In pgAdmin III create a superuser with the name "sde" and the password "sde".
8. Open ArcMap or ArcCatalog and run the "Create Enterprise Geodatabase" geoprocessing tool (Tools – Data Management Tools – Geodatabase Administration) from ArcGIS for Desktop.
9. Choose PostgreSQL, enter the instance URL.
10. Enter optionally the database administrator account (e.g.: postgres) and also the geodatabase account (e.g.: sde)
11. Select the "Authorization file" which is stored on the server (C:\Program Files\ESRI\License10.2\sysgen)
12. Click OK to run the tool.
13. If the tool was successfully performed check the new database in pgAdmin III.

Registering the spatial ArcSDE database with the server

The non-spatial simSALUD database and the spatial sde database must be registered to join both databases for the visualization page on the simSALUD web application.

1. Open the "Catalog Tree" in ArcCatalog or ArcMap
2. Right click on the previously created ArcGIS server (if not connected - click on connect) and click on "Server Properties".
3. Click on the tab "Data store" and click on the black cross to register a new database and give it a name "myGeodatabase".
4. Click on Add and enter all required "PostgreSQL" parameters with the instance: "simsalud.cti.ac.at". In terms of importing spatial data to the database it is recommend using the "sde" superuser and the password "sde". Save user name and password. Choose the existing created "sde" database.
5. Click on OK

Registering the simsalud PostgreSQL database

This spatial database stores all spatial data such as district, municipality for visualizing on the simSALUD web application.

1. Open the "Catalog Tree" in ArcCatalog or ArcMap
2. Right click on the previously created ArcGIS server (if not connected - click on connect) and click on "Server Properties".
3. Click on the tab "Data store" and click on the black cross to register a new database and give it a name "NonSpatial".
4. Click on Add and enter all required "PostgreSQL" parameters with the instance: "simsalud.cti.ac.at". In terms of importing spatial data to the database it is recommend using as user name "postgres" and the password "postgres". Save user name and password. Choose the previously created "simsalud" database.
5. Click on OK

Adding a new shape file to the simSALUD web application

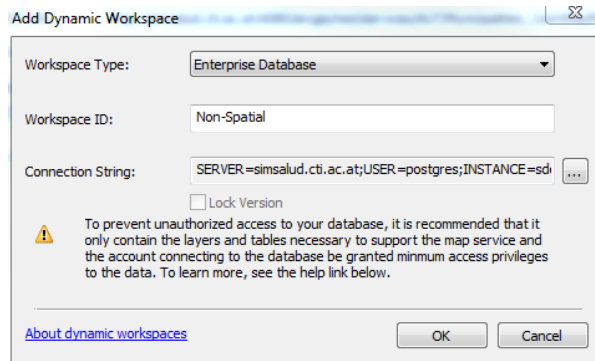
1. Open ArcMap
2. Open the ArcCatalog Tree and right click on "Database Connections" – "Add Database Connection" and enter the same values as previously entered in: Registering the spatial ArcSDE

(Publishing a service)

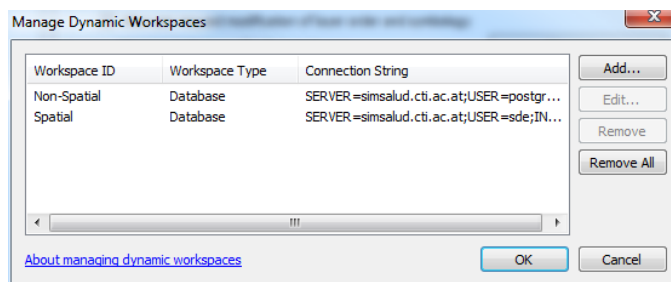
- database with the server.
3. Click on the new created “myGeodatabase” in Database Connection.
 4. Click “Import” – “Feature Class (single)” and choose the file which should be imported.
 5. After the import drag the file from the geodatabase to the “Table of Contents” in ArcMap.
 6. Open the “Attribute table” of the feature class and check if the field which must be joined to the simulated data (same values as “region_id” column of each census file) is from type “String”. Right click on the field and “Properties” and check the “Type”. If it is not a “String” type than create a new field: Click on the arrow of the first icon in the “Table” view and click “Add field...”. Name it (e.g.: region_id_join) and choose “String” as data type. (If the edit mode is locked than you have to stop the service at first where the data is published). With Field Calculator copy the old “region_id” values to the new field to become these value to an “String” data type.
 7. For publishing the service click: File – Share As... – Service....
 8. Select “Publish a service” and click “Next”.
 9. Enter the Service name (e.g.: Districts_Austria)
 10. Either enter a new folder name or add the service to an existing folder
 11. Click “Analyze”.
 12. Only three warning messages (no errors) should appear. These can be ignored → Click “Publish”.

Joining both databases

1. These steps must be done for each published service to join the simulated data with the spatial data.
2. Open the “Catalog Tree” in ArcMap or ArcCatalog
3. Go to “GIS Servers” and navigate to the service which was uploaded just before.
4. Right click on the service and click “Service Properties...”.
5. Navigate to “Mapping”
6. Check the “Allow per request modification of layer order and symbology”
7. Click Manage...
8. Click Add
9. Register the “nonSpatial” database
10. Enter the name “Non-Spatial” for the workspace ID
11. Select the Connection String
 - a. EITHER Select the appropriate database connection OR
 - b. Select the connection String: The default location for the file is:
\\<computer_name>\Users\<user_name>
\AppData\Roaming\ESRI\Desktop<release#>\ArcCatalog



12. Click OK
13. Click Add
14. Register the “Spatial” database
15. Enter the name “Non-Spatial” for the workspace ID
16. Select the Connection String
 - a. EITHER Select the appropriate database connection OR
 - b. Select the connection String: The default location for the file is:
`\\<computer_name>\Users\<user_name>\AppData\Roaming\ESRI\Desktop<release#>\ArcCatalog`



17. Click OK

Add the required map parameters in eclipse

The following steps must be added / changed in eclipse enable the visualization functionality.

1. Go to eclipse and open the framework simsalud
2. Open the file: globalVariableAndFunctions.js
3. (optionally) add the new folder name with “slash” into the “arcGISMapServiceNames” array. This is used to hide all other system folder on the simsalud visualization page
4. change the name of the service URL
5. Open the file: ModeUtile.java
6. Scroll down to the Visualization Parameters
7. Add the arcgisServiceName: Parameter can be found here: Open Catalog Tree in ArcMap – Right click on the service – Properties - Name
8. Add the arcgisServiceNameTableName: Paramter can be found here: Open Catalog Tree in ArcMap – Right click on the service – Service Workspaces – Referenced. Needed are the name of the database (sde), the name of the schema in postgres (sde) and the name of the table, e.g.: “sde.sde.districts_austria”.
9. Add the arcgisServiceNameJoinColumnString: This is the name of the field which was created as “String” datatype before publishing the service. Used to join the spatial published data with the simulated data.

10. Open the file: mapProvider.jsp
11. Copy and paste one else if block below the previous one
12. Change the parameter highlight in bold and yellow:

```
} else if (arcgisServerGeoDataSelect == "AUT/Districts_Austria") {  
    //define the layer's data source from a table  
    var dataSource = new TableDataSource();  
    dataSource.workspaceId = '<%=ModelUtile.workspaceID%>';  
    dataSource.dataSourceName = "sde.sde.districts_austria";  
    var layerSource = new LayerDataSource();  
    layerSource.dataSource = dataSource;  
  
    var popupTemplate = new PopupTemplate({  
        title: "{bezirk}",  
        fieldInfos: [{  
            fieldName: "bezirk",  
            visible: true,  
            label: "District"  
        }],  
    });  
}
```

Hint: In eclipse (developing the application) it is not possible to download simulated files via the simulation page. Simulated and validated files are stored here: "C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\simisalud\data\user_... ". For using the application the following steps are required:

Use simSALUD on localhost

Do the next step only if the simSALUD web application will be used without developing anything. This requires to have an workable Tomcat 7 server [see *Install Apache Tomcat 7*] and a PostgreSQL database [see **Error! Reference source not found.**] with the required structure for using simsalud [see *Configure PostgreSQL*]

1. Right click on simsalud in "Project Explorer" – Export – WAR file.
2. Select a destination, name the file "simsalud" and check "Export source files" and "Overwriting existing file" - Finish
3. Close eclipse and start "Monitor Tomcat" (Start button – Apache Tomcat 7.0 Tomcat 7 folder)
4. Click the "Start" button to start the Tomcat server
5. Open your browser and type in: "localhost:8080"
6. Click the "Manager App" button and enter the username and password (e.g.: tomcat).
7. Scroll down to "Deploy – War file to deploy", click on "Browse" to select the "simsalud.war" file and click "Deploy".
8. Click on the new created application "simsalud" to open the application.
9. Run the simulation and download your results

Hint: If you have already deployed simsalud.war, at first you must "Undeploy" the old version. If this does not work: Stop the server and navigate to the installation directory of the server (folder webapps) and delete the simsalud folder. Restart the server and proceed with step 5.

Error handling

Message: Wrong file format!
Please use *.csv!

Message: You have no access
writes to copy files on your
machine!

You don't see all published
features on the map.

Any error messages are either displayed on any simSALUD pages or a more detailed info is displayed in the console of eclipse

Possible Solution: Check if the data format csv is installed on your local machine (e.g.: MS Office, OpenOffice)

Possible solution: Check if you have administrator rights and if you have started eclipse in administrator mode.

Increase the maximum number of records in the service properties for each published service

Appendix 1: Database Structure

Table: sim_columnnames

Table: sim_surveycensusmatch

Table: sim_filecolumnjoin

Table: sim_filenames

Table: sim_filecolumnjoin_output

Table: sim_columnnames_output

- The simSALUD application is based on a PostgreSQL database to store all input data files as well as all simulated and validated data.
- Each time a user opens the web application and starts to simulate data, a new schema is created within the database *simсалud*. The schema name consists of the name "user_" the date and time of the user and can so be seen as a unique timestamp for each user.
- After uploading the first survey file the application creates a database structure consisting of six tables:

Stores for each survey and census files all original column names (.csv files) and a corresponding database column name (avoiding umlauts)

Stores primary keys for all survey and census data to match these data (Survey/Census Match Tab within the simSALUD application)

Connect the table between "sim_columnnames" and "sim_filenames".

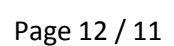
Stores the original filename and filetypes of all uploaded files and a corresponding tablename (avoiding umlauts)

Connect the table between "sim_filenames" and "sim_columnnames_output".

Stores all additional output columnnames

- The uploaded survey file is copied to the server and the data is stored in an "input_[surveyFileName]" table.
- Also all uploaded census files are copied to the server and stored in separate "input_[censusFile(s)name]" table.
- After running the simulation, for each census table a new "adjust_[censusFile(s)name]" is created. For a detailed description of the adjusting algorithm see the tutorial of the simSALUD application.

simSALUD setup with visualization



Appendix 2: Java packages

- In Eclipse the simSALUD simulation is defined as a dynamic web project where all java classes are stored in the “Java Resources” folder and all client side files are stored in the “WebContent” folder.
- All classes are stored into six different packages. For a detailed description of all classes check the class and method description in the files.

Package: at.fh.kaernten.action

The *action package* contains classes to handle the upload of the survey and census files, the *reweighted class* including all algorithms and the *simulation class* which is executed for the simulation run.

Package: at.fh. kaernten.database

The *database package* contains the classes to define all database connection settings and the class with all needed database queries.

Package: at.fh. kaernten.export

The export package contains the classes to generate the output .csv files and to zip the files for downloading.

Package: at.fh. kaernten.utile

The utile package contains a class which provides all settings parameter for the simulation such as the output path, name of the zip file and so on.

Package: at.fh. kaernten.validation

The validation package contains the classes to execute the volition as well as all provide validation algorithms.