Maverig Documentation

Release 1.0.5

PG-Maverig

Feb 27, 2018

Contents

1 Quickstart **3**

2 Installation **5**
2.1 Linux 5
2.2 Windows 7

3 User Manual **11**

4 Project Documentation **13**

5 Source Documentation **15**
5.1 maverig.data 15
5.2 maverig.models 19
5.3 maverig.presenter 28
5.4 maverig.views 42
5.5 maverig.utils 55
5.6 maverig.tests 63
5.7 `maverig.EntryPoint` 73
5.8 `maverig.demo` 73

6 About Maverig **75**
6.1 Contributors 75

7 Indices and tables **77**

Python Module Index **79**

Contents:

CHAPTER 1

Quickstart

Maverig runs on the operation systems Linux, OSX and Windows. It requires Python 3.4 or higher and the package manager pip.

Maverig can simply be installed with the command:

```
$ pip install maverig
```

However, you need to install some requirements beforehand:

```
PySide==1.2.2
numpy>=1.8.1
networkx>=1.8.1
python-dateutil>=2.2
matplotlib>=1.4.2
colormath>=2.0.2
polib>=1.0.6
pyzmq>=14.3.1
mosaik>=2.1.2
mosaik-api>=2.1
mosaik-pypower>=0.7
```

The installation of *mosaik*, *mosaik-api* and *mosaik_pypower* is described in detail in the mosaik installation guide. There is a detailed *installation guide for Linux* (Ubuntu) and some hints for *installation under Windows*. A detailed guide for Windows and OSX will be added soon.

CHAPTER 2

Installation

This guide contains detailed installation instructions for *Linux* and *Windows*. A detailed guide for OSX will be added soon. If you're not proficient with installation procedures in Python, we suggest that you stick close to the installation guide.

Linux

This guide is based on *(K)ubuntu 14.04 Trusty Tahr, 64bit*. It works also on *(K)ubuntu 16.04 Xenial Xerus, 64bit*, the only difference is the installation of PySide which is a little bit tricky in both cases.

Maverig comes with a demo scenario which can be used to check if the installation is working correctly. The maverig demo requires some simulators included in mosaik-demo. For that we install mosaik-demo first.

As common in Python we use virtualenv to create virtual Python environments to keep the dependencies required by different projects in separate places. You can find more information in this guide to virtualenv. We will install maverig and mosaik-demo into a virtualenv called `maverig`. We also need a folder where we put maverig and mosaik-demo. Put these folders were you usually keep your source code. In the following we assume that this is a folder called `~/Code/`.

Installing mosaik-demo

1. Maverig and the mosaik-demo require Python >= 3.3. Ubuntu ships with Python 3.4, so this is okay. We also need pip, a package manager for Python packages and *virtualenv* as mentioned above:

```
$ wget https://bootstrap.pypa.io/get-pip.py
$ sudo python get-pip.py
$ sudo pip install -U virtualenv
```

2. Mosaik-demo requires the libraries *NumPy*, *SciPy* and *h5py*. We also need to install the version control tool *Mercurial*. You can use the packages shipped with Ubuntu. We use **`apt-get`** to install NumPy, SciPy, and h5py as well as Mercurial.

```
$ sudo apt-get install mercurial python3-numpy python3-scipy python3-h5py
```

3. Now we need to create and activate a virtual environment for maverig and its dependencies. The common location for virtualenvs is `~/.virtualenvs/`:

```
$ virtualenv -p /usr/bin/python3 --system-site-packages ~/.virtualenvs/maverig
$ source ~/.virtualenvs/maverig/bin/activate
```

Your command line prompt should now start with `(maverig)` and roughly look like this: `(maverig)user@kubuntu:~$`. The flag *–system-site-packages* makes sure that the packages installed with **`apt-get`** can be used inside the virtualenv. The *-p* option makes sure that a Python3.4 interpreter is used inside the virtualenv. Without it the standard interpreter, which is Python2.7 in Ubuntu 14.04, would be used.

4. You can now get the mosaik-demo repository by cloning it into a folder where you store all your code and repositories. As mentioned above we'll use `~/Code/`:

```
(mosaik)$ mkdir ~/Code
(mosaik)$ hg clone https://bitbucket.org/mosaik/mosaik-demo ~/Code/mosaik-demo
```

5. Now we need to install all requirements (mosaik and the simulators) for mosaik-demo and can finally run the demo:

```
(mosaik)$ cd ~/Code/mosaik-demo/
(mosaik)$ pip install -r requirements.txt
(mosaik)$ python demo.py
```

If no errors occur, the last command will start the demo. The web visualisation shows the demo in your browser at http://localhost:8000. You can click the nodes of the topology graph to show a time series of their values. You can also drag them around to rearrange them.

You can cancel the simulation by pressing `Ctrl-C`.

Installing maverig

1a. This point is only for (K)Ubuntu 14.04. For (K)Ubuntu 16.04 refer to 1b. Maverig requires PySide, a Python binding for Qt. Unfortunately the required version of PySide (1.2.2) is not available for Ubunutu 14.04 via **`apt-get`** so we have to build it ‘‘manually´´. The build requires a number of prerequisits that we can install using **`apt-get`**:

```
(maverig)$ sudo apt-get install python3-dev build-essential cmake libqt4-dev libphonon-dev pythc
```

Now we can build PySide in our virtualenv.

```
(maverig)$ pip install PySide==1.2.2
```

The build requires some time, don't get impatient.

For some reason PySide's post install script is not executed properly, so we have to do it manually. If you installed everything on its default locations, as described above, you execute the script as shown below, else you have to adapt your path.

```
(maverig)$ python ~/.virtualenvs/maverig/bin/pyside_postinstall.py -install
```

1b. **This point is only for (K)Ubuntu 16.04. For (K)Ubuntu 14.04 refer to 1a.**

Maverig requires PySide, a Python binding for Qt. The required version of PySide (1.2.2) is available for Ubunutu 14.04 via **`apt-get`**:

```
$ sudo apt-get install python3-pyside
```

For some reason `egg-info` isn't installed properly so that pip cannot find the installed PySide version. We have to install the `egg-info` manually. We find it in the PySide distribution.

```
$ cd ~/Code
$ wget https://pypi.python.org/packages/source/P/PySide/PySide-1.2.2.tar.gz
```

Extract the source distribution

```
$ tar -xvzf PySide-1.2.2.tar.gz
```

Now we have to copy the egg-info into the site-distribtution folder. If you followed the standard rules and everything is installed in default locations you can use the following command, otherwise you have to adapt your path.

```
$ sudo cp -r ~/Code/PySide-1.2.2/pyside_package/PySide.egg-info /usr/lib/python3/dist-packages/
```

2. From this step on the guide is for both Ubuntu versions, 14.04 and 16.04. Next step is to get the maverig source code and to install the remaining requirements.

```
(maverig)$ hg clone https://bitbucket.org/mosaik/maverig ~/Code/maverig
(maverig)$ cd ~/Code/maverig
(maverig)$ pip install -r requirements.txt
```

3. After all these preparations we are finally able to install maverig itself.

```
(maverig)$ pip install maverig
```

4. If no error occured we can now start maverig.

```
(maverig)$ maverig
```

You'll find a demo scenario in maverig (file|open, go to folder scenarios, select demo.mvrg). If you open it and start the simulation (simulation|run or F5) you see the animation. You can click on any element to see further information about it on the left side. For the use of maverig consult the *manual*.

Windows

This guide is based on *Windows 7 Professional 64 bit*.

Maverig comes with a demo scenario which can be used to check if the installation is working correctly. The maverig demo requires some simulators included in mosaik-demo. For this, we install mosaik-demo first.

As common in Python we use virtualenv to create virtual Python environments to keep the dependencies required by different projects in separate places. You can find more information in this guide to virtualenv. We will install maverig and mosaik-demo into a virtualenv called `maverig`. We also need a folder where we put maverig and mosaik-demo. Put these folders where you usually keep your source code. In the following we assume that this is a folder called `Code` that is located in your user directory.

Installing mosaik demo

1. Mosaik and the demo scenario require Python 3.4. You can find the Windows installer here. Select the correct Python 3.4 version for your operating system (32 or 64 Bit).
   (a) When the download finished, double-click the installer.
   (b) Select *Install for all users* and click *Next* >.
   (c) The default installation path is okay. Make sure that the installation path does not contain any blanks. Click *Next* >.
   (d) In the *Customize Python* page, click on the *Python* node and select *Entire feature will be installed on local hard drive*. Make sure that *Add python.exe to Path* is enabled. Click *Next* >.
   (e) When Windows asks you to allow the installation, allow the installation. Wait. Click *Finish*.

   This also installs the Python package manager pip.

2. Download and install Mercurial. We recommend to use TortoiseHg. Download the correct version for your operation system and follow the installation assistant.

3. The version of pip that comes with Python 3.4 is quite old, so we have to update it. Open a terminal window: Press the `Windows` key (or click on the start menu) and enter `cmd`. Your terminal prompt should look like `C:\Users\yourname>`.

   Upgrade pip by:

```
C:\Users\yourname>python -m pip install --upgrade pip
```

   Make sure that Python and Mercurial is installed correctly by checking its versions.

```
C:\Users\yourname> python --version
C:\Users\yourname> pip --version
C:\Users\yourname> hg --version
```

   Depending on the versions you installed your console window should show something like this:

```
C:\Users\yourname>python --version
Python 3.4.4

C:\Users\yourname>pip --version
pip 9.0.1 from c:\python34\lib\site-packages (python 3.4)

C:\Users\fschloegl>hg --version
Mercurial Distributed SCM (version 4.2.1)
(see https://mercurial-scm.org for more information)

Copyright (C) 2005-2017 Matt Mackall and others
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

   If you get error messages the program(s) is(are) not installed correctly. In most cases the environment variable PATH is not set correctly.

4. As mentioned above we need to create a virtualenv called `maverig`. The common location for virtualenvs is under `Envs/` in your users directory:

```
C:\Users\yourname> pip install -U virtualenv
C:\Users\yourname> virtualenv -p c:\python34\python.exe Envs\maverig
C:\Users\yourname> Envs\maverig\Scripts\activate.bat
```

   You can drop the command line argument **`-p`** if you have only one installed version of Python. If you have several versions of Python the path after **`-p`** has to point at the respective Python executable.

   Your command line prompt should now start with "(maverig)" and roughly look like this: `(maverig) C:\Users\yourname>`.

Note: If your Windows account type is *Standard User*, you need to open the terminal with administarator privileges (right-click the Terminal icon, then *open as Administrator*). Make then sure that you are in your user directory:

```
C:\Windows\system32> cd \Users\yourname
C:\Users\yourname>
```

---

5. Mosaik uses PYPOWER as grid simulator which requires *NumPy* and *SciPy*. Further we need *h5py* for the database adapter of the demo.

Christoph Gohlke provides installers for them (NumPy, SciPy, h5py). Select the appropriate files for your Python installation (32bit or 64bit, Python version), e.g. for Windows 64bit and Python 3.4: *numpy-1.11.3+mkl-cp34-cp34m-win_amd64.whl*, *scipy-0.19.1-cp34-cp34m-win_amd64.whl*, *h5py-2.7.0-cp34-cp34m-win_amd64.whl*.

Use a numpy-version <= 1.11! Numpy 1.12 or 1.13 don't work.

Note: Run `python -c "import sys; print(sys.version)"` from the command prompt in order to get the system architecture and Python version.

If you have a 64bit Windows, but installed a 32bit Python, also use the 32bit versions of NumPy etc.

Download them into your downloads folder and install them via the following commands:

```
(maverig) C:\Users\yourname> pip install Downloads\numpy-1.11.3+mkl-cp34-cp34m-win_amd64.whl
(maverig) C:\Users\yourname> pip install Downloads\scipy-0.19.1-cp34-cp34m-win_amd64.whl
(maverig) C:\Users\yourname> pip install Downloads\h5py-2.7.0-cp34-cp34m-win_amd64.whl
```

6. Create a folder for your code (if none exists yet) and clone the mosaik-demo repository into it:

```
(maverig)C:\Users\yourname> mkdir Code
(maverig)C:\Users\yourname> cd Code
(maverig)C:\Users\yourname\Code> hg clone https://bitbucket.org/mosaik/mosaik-demo
```

7. Now we only need to install all requirements (mosaik and the simulators) and can finally run the demo:

```
(maverig)C:\Users\yourname\Code> cd mosaik-demo
(maverig)C:\Users\yourname\Code\mosaik-demo> pip install -r requirements.txt
(maverig)C:\Users\yourname\Code\mosaik-demo> python demo.py
```

If no errors occur, the last command will start the demo. The web visualisation shows the demo in your browser at http://localhost:8000. You can click the nodes of the topology graph to show a time series of their values. You can also drag them around to rearrange them.

You can cancel the simulation by pressing `Ctrl-C`.

Installing maverig

1. Clone the maverig repository into your `Code` folder:

```
(maverig)C:\Users\yourname\Code\mosaik-demo> cd C:\Users\yourname\Code
(maverig)C:\Users\yourname\Code> hg clone https://bitbucket.org/mosaik/maverig
```

2. Install all requirements of maverig:

```
(maverig)C:\Users\yourname\Code> cd maverig
(maverig)C:\Users\yourname\Code\maverig> pip install -r requirements.txt
```

3. After all these preparations we are finally able to install maverig itself.

```
(maverig)C:\Users\yourname\Code\maverig> pip install maverig
```

4. If no error occured we can now start maverig.

```
(maverig)$ C:\Users\yourname\Code\maverig> cd maverig
(maverig)$ C:\Users\yourname\Code\maverig> python EntryPoint.py
```

You'll find a demo scenario in maverig (file|open, go to folder scenarios, select demo.mvrg). If you open it and start the simulation (simulation|run or F5) you see the animation. You can click on any element to see further information about it on the left side. For the use of maverig consult the *manual*.

CHAPTER 3

User Manual

You can download the User Manual in the following languages:

- `english - User Manual (PDF)`
- `deutsch - Benutzerhandbuch (PDF)`

CHAPTER 4

Project Documentation

Here you can download the Project Documentation of the **Maverig Project** (April 2014 - March 2015) in German language.

The Project Documentation describes the complete software development process of Maverig including the following parts:

- Project Organization
- Smart Grid Basics
- Maverig Requirements
- Software Concepts and Design
- Implementation Documentation
- Validation
- Outlook

Download (PDF): `deutsch - Projektdokumentation`

CHAPTER 5

Source Documentation

The Source Documentation describes classes and methods in Maverig source code at https://bitbucket.org/Sash221/maverig/src.

This may help you to understand the software in detail or if you plan further development of additional features or component descriptions.

maverig.data

maverig.data.settings

- *maverig.data.settings.abstractSettings*
- *maverig.data.settings.defaultSettings*
- *maverig.data.settings.heatValueEffect*
- *maverig.data.settings.settings*

maverig.data.settings.abstractSettings

class maverig.data.settings.abstractSettings.**Tab**
Bases: object

class maverig.data.settings.abstractSettings.**Setting**
Bases: object

class maverig.data.settings.abstractSettings.**CheckBoxSetting**
Bases: *maverig.data.settings.abstractSettings.Setting*

class maverig.data.settings.abstractSettings.**InputSetting**
Bases: *maverig.data.settings.abstractSettings.Setting*

class maverig.data.settings.abstractSettings.**ComboBoxSetting**
Bases: *maverig.data.settings.abstractSettings.Setting*

maverig.data.settings.defaultSettings

maverig.data.settings.heatValueEffect

class maverig.data.settings.heatValueEffect.**HeatValueEffectKeys**
Bases: object

EFFECT_BAR = 'Bar'

EFFECT_BLUR = 'Blur'

EFFECT_COLOR = 'Color'

EFFECT_SHADOW = 'Shadow'

EFFECT_TRANSPARENCY = 'Transparency'

maverig.data.settings.settings

class maverig.data.settings.settings.**SettingTypes**
Bases: object

COMBO_BOX = 'combo_box'

INPUT = 'input'

CHECK_BOX = 'check_box'

class maverig.data.settings.settings.**Settings**
Bases: object

tabs = [<maverig.data.settings.abstractSettings.Tab object at 0x7f0a3087c588>, <maverig.data.settings.abstractSettings.

general_settings_tab = <maverig.data.settings.abstractSettings.Tab object>

languages = <maverig.data.settings.abstractSettings.ComboBoxSetting object>

simulation_settings_tab = <maverig.data.settings.abstractSettings.Tab object>

day_night_vis_setting = <maverig.data.settings.abstractSettings.CheckBoxSetting object>

heat_value_effect_setting1 = <maverig.data.settings.abstractSettings.CheckBoxSetting object>

heat_value_effect_setting2 = <maverig.data.settings.abstractSettings.CheckBoxSetting object>

- *maverig.data.config*
- *maverig.data.dataHandler*

maverig.data.config

maverig.data.config.**import_method**(*module_method_address*)
Return the python method at the given address.

module_method_address is a string which consists of module address and method name separated by ':'.

maverig.data.config.**read_components**()
Return all component descriptions from *maverig/data/components* as dict from *sim_model* to component description content

maverig.data.config.**read_simulators**()
Return all simulator descriptions as dict from simulators *name* to simulator description content

maverig.data.config.**read_json**(*filename*)
Return the content dict of a *.json file

maverig.data.config.**write_json**(*filename*, *obj*)
Write a content dict (*obj*) into a *.json file

maverig.data.config.**read_config**()
Return the application configuration dict from *maverig/data/cfg.json*

maverig.data.config.**write_config**(*cfg*)
Write *cfg* into the configuration file *maverig/data/cfg.json*.

maverig.data.config.**restore_config**()
Write the default settings in `maverig.data.settings.defaultSettings` into the configuration file *maverig/data/cfg.json*.

class maverig.data.config.**ConfigKeys**
Bases: `object`

Constants with configuration dict keys.

UI_STATE = 'ui_state'

MAIN_WINDOW_GEOMETRY = 'main_window_geometry'

MAIN_WINDOW_STATE = 'main_window_state'

SPLITTER_MAIN_GEOMETRY = 'splitter_main_geometry'

SPLITTER_MAIN_STATE = 'splitter_main_state'

SPLITTER_LEFT_GEOMETRY = 'splitter_left_geometry'

SPLITTER_LEFT_STATE = 'splitter_left_state'

SPLITTER_RIGHT_GEOMETRY = 'splitter_right_geometry'

SPLITTER_RIGHT_STATE = 'splitter_right_state'

IS_ATTRIBUTE_PANEL_VISIBLE = 'is_attribute_panel_visible'

IS_COMPONENT_PANEL_VISIBLE = 'is_component_panel_visible'

IS_CONSOLE_PANEL_VISIBLE = 'is_console_panel_visible'

IS_PROGRESS_BAR_VISIBLE = 'is_progress_bar_visible'

IS_PROPERTY_PANEL_VISIBLE = 'is_property_panel_visible'

IS_STATUS_BAR_VISIBLE = 'is_status_bar_visible'

ATTRIBUTE_GRAPHS_VISIBLE = 'attribute_graphs_visible'

GENERAL_SETTINGS = 'general_settings'

LANGUAGE = 'language'

SIMULATION_SETTINGS = 'simulation_settings'

IS_DAY_NIGHT_VIS_ENABLED = 'is_day_night_vis_enabled'

IS_HEAT_VALUE_EFFECT_FOR_GRIDS_ENABLED = 'is_heat_value_effect_for_grids_enabled'

IS_HEAT_VALUE_EFFECT_FOR_CPP_ENABLED = 'is_heat_value_effect_for_cpp_enabled'

HEAT_VALUE_EFFECT_GRIDS = 'heat_value_effect_grids'

HEAT_VALUE_EFFECT_CPP = 'heat_value_effect_cpp'

MODE_PANEL_SETTINGS = 'mode_panel_settings'

INVISIBLE_COMPONENTS = 'invisible_components'

SHOW_INVISIBLE_COMPONENTS = 'show_invisible_components'

maverig.data.config.**raster_pos**(*pos*)
Return the nearest PySide.QtCore.QPointF raster coord position to pos.

maverig.data.config.**ACTIVATED_COMPONENT_MODE**()

maverig.data.config.**ACTIVATED_HAND_MODE**()

maverig.data.config.**ACTIVATED_SELECTION_MODE**()

maverig.data.config.**ACTIVATED_SIMULATION_MODE**()

maverig.data.config.**ACTIVATED_AUTO_LAYOUT_MODE**()

maverig.data.config.**SEPARATOR**()

maverig.data.config.**SIMULATION_SPEED**()

maverig.data.config.**SIMULATION_PAUSED**()

maverig.data.config.**CREATION_INVALID**()

maverig.data.config.**DOCKING_VALID**()

maverig.data.config.**LINE_TOO_SHORT**()

maverig.data.config.**DOCKING_INVALID**()

maverig.data.config.**DOCKING_NO_ITEMS**()

maverig.data.config.**ZERO_KM_LENGTH**()

maverig.data.config.**FILE_SAVED**()

maverig.data.config.**FILE_OPENED**()

maverig.data.config.**MULTI_SELECT_DIFFERENT_VALUES**()

maverig.data.config.**SIMULATION_COMPLETED**()

maverig.data.config.**create_components_language_po_entries**(*components=None*)
Write keys found in components into language *.po files under *maverig/data/languages* in order to simplify the component developers translation process.

maverig.data.dataHandler

maverig.data.dataHandler.**ensure_dir**(*path*)
Create directories contained in path if they don't exist.

maverig.data.dataHandler.**get_maverig_dir**()
Return the maverig directory, e.g. `'C:\Programs\maverig\maverig'`.

maverig.data.dataHandler.**get_relpath**(*path*)
Return the path relative to maverig directory.

For example `get_relpath('C:\Programs\maverig\maverig\data')` will return `'maverig\data'`.

maverig.data.dataHandler.**get_normpath**(*path*, *sub_dir=''*, *create_dir=False*)
Return the complete normalized path for the current os environment.

For example `get_normpath('maverig\data\configs\cfg.json')` or `get_normpath('cfg.json', sub_dir='maverig\data\configs')` will return `'C:\Programs\maverig\maverig\data\configs\cfg.json'`.

Set *create_dir* to True if non existing directories should be created.

maverig.data.dataHandler.**get_component_icon**(*filename*)
Return the complete component icon path for the given *filename* relative to *maverig/data/components/icons*.

maverig.data.dataHandler.**get_icon**(*filename*)
Return the complete icon path for the given filename relative to *maverig/data/icons*.

maverig.data.dataHandler.**get_lang_path**(*create_dir=False*)
Return the complete languages locale path of *maverig/data/languages/*.

maverig.data.dataHandler.**get_temp_file**(*filename*)
Return the complete temporary files path for the given filename relative to *maverig/data/temp*.

maverig.data.dataHandler.**get_config_file**(*filename*)
Return the complete configuration file path for the given filename relative to *maverig/data/configs*.

maverig.models

- *maverig.models.model*
- *maverig.models.modelGraph*
- *maverig.models.modelSimulation*

maverig.models.model

maverig.models.model.**fast_deepcopy**(*x*)
Return a deepcopy of *x*.

x may be or contain the following types:

```
{dict, set, list, tuple, datetime, string, int, float, bool, NoneType}
```

class maverig.models.model.**ElemPort**(*elem_id*, *port*)
Bases: `tuple`

An element port instance (e.g. `ElemPort('CSV.House_1', '0')`) describes a specific `port` (e.g. `'0'`) of an element specified by `elem_id` (e.g. `'CSV.House_1'`), which may be docked to other element ports.

Element ports are often abbreviated with `ep`.

In all model methods, element ports can also be passed as list (e.g. `['CSV.House_1', '0']` and will be automatically converted to *ElemPort* when needed.

elem_id
Alias for field number 0

port
Alias for field number 1

class maverig.models.model.**ProgramMode**
Bases: object

Represents the main program mode: composition, simulation, or simulation paused.

composition = 'composition mode'
Composition mode, where scenario can be edited.

simulation = 'simulation mode'
Simulation mode, where simulation or recorded simulation history "video" is running.

simulation_paused = 'simulation paused'
Simulation paused mode, where simulation is paused.

class maverig.models.model.**Mode**
Bases: object

Represents the currently selected mode for the composition.

selection = 'selection mode'
In selection mode, all elements may be moved and edited.

hand = 'hand mode'
In hand mode, the complete scenario is movable and elements can't be edited or created.

comp = 'component mode'
In component mode, only elements of a the selected component in mode panel may be edited and created.

sim = 'simulation mode'
In simulation mode, the scenario can only be watched and elements can't be edited. Similar to hand mode but may differ in element visualizations.

class maverig.models.model.**Model**
Bases: object

Model manages the complete scenario state: - simulation start and end time - elements as component instances and their dockings - clipboard - selected elements - modi (comp_mode, selection_mode, hand_mode) and - events.

auto_update_components_languages = None
Switch for automatic creation of *.po language entries for new component texts.

elements = None
A dict mapping elem_id to element instance." Elements instances represent a Power or Logic Unit (here a PQBus example) with customizable data for the following purposes:

•link to component description by `model.get_component(elem_id)` or `model.components[sim_model]`

(here in maverig/data/components/PyPower.PQBus.json)

•reference data (*elem_id*)

•custom visualization (*icon*),

•positioning for each port (*pos*),

•dockings to/from other elements and their ports (*out*, *in*)

•data for simulation runtime and initialization:

– parameters and their values (*params*)

– attributes at simulation runtime with their static or listed dynamic values (*attrs*)

- reference to mosaik entity (*mosaik_full_id*)

  mosaik_full_id and *attrs* will be automatically filled during simulation time

Example elements dictionary with PyPower PQBus instance:

```
{
    "PyPower.PQBus_4": {
        "sim_model": "PyPower.PQBus",
        "icon": "bus.svg",
        "elem_id": "PyPower.PQBus_4",
        "mosaik_full_id": "PyPower-0.0-PyPower.PQBus_4",
        "docking_ports": {
            "0": {
                "pos": [450.0,200.0]
                "out": [],
                "in": [["PyPower.Branch_3","1"], ["PyPower.Branch_5","0"], ["CSV.House_4","1
            }
        },
        "params": {
            "bus_type": "PQ",
            "base_kv": 0.23,
            "fbus": null,
            "tbus": null
        },
        "attrs": {
            "Vl": 230.0,
            "Q": [0.0, 0.0, 0.0],
            "Vm": [230.14789278916751, 230.23588331696217, 230.22035839218473],
            "P": [381.21, 174.35, 116.43999999999998],
            "Va": [-0.17130258343689386, -0.17013456872934510, -0.16981416413842957]
        }
    },
    "CSV.House_1": {...}
}
```

history_undo = None
The history list of scenarios that can be undone. `history_undo[-1]` is the nearest to the current scenario.

history_redo = None
The history list of scenarios that can be redone. `history_redo[-1]` is the nearest to the current scenario.

tmp_scenario_copy = None
A copy of the current scenario since the last history relevant change occured.

saved_scenario_copy = None
A copy of the last saved scenario. Use `self.saved_scenario_copy == self.tmp_scenario_copy` comparison to check whether scenario has changed since last save.

uid = None
A dict mapping sim_model to current id counter for creating elements.

clipboard_elements = None
A dict mapping elem_id to element instance containing all the elements which are currently in clipboard.

graph
The *maverig.models.modelGraph.ModelGraph* as networkx-Graph representation of the scenario.

simulation
The *maverig.models.modelSimulation.SimulationServer* which manages the simulation.

components
A dict mapping *sim_model* to component description. Component descriptions are read from JSON-Files in *maverig/data/components*.

simulators
A dict mapping simulator name to simulator description. Simulator descriptions are read from JSON-Files in *maverig/data/components/simulators*.

sim_start
The simulation start time as `datetime.datetime`.

sim_end
The simulation end time as `datetime.datetime`.

sim_step_size
The step size of the simulation in seconds. Standard is 1800 (= 30 minutes).

sim_index
The current simulation time index, which is used to address the *current timestamp* and the *current dynamic attribute value*.

Setting it will result in the closest valid index if it exceeds the boundaries of collected simulation timestamps.

sim_end_index
The last possible simulation time index of an completed simulation.

vid_speed
Simulation visualization speed interval. The interval in milliseconds, how much time to pass until showing the next simulation timestep.

Setting this property will result in the closest supported speed interval.

Supported intervals (slow → fast):

```
2000, 1750, 1500, 1250, 1000, 750, 500, 250, 50
```

vid_speed_rel
Simulation visualization speed factor. Note that this property does not represent an exact speed factor.

Setting this property will result in the closest supported speed factor.

Supported factors (slow → fast):

```
0.5, 0.625, 0.75, 0.875, 1, 1.5, 2, 4, 8
```

sim_progress
The running simulation progress in percent (`0..100`).

sim_timestamp
The current time stamp as `datetime.datetime` of the current simulation time index.

Setting this property will result in the closest collected simulation time stamp.

duration
The simulation duration in seconds calculated by *sim_start* and *sim_end* time difference.

language
The currently active installed language for internationalization (e.g. `'en_EN'`).

program_mode
The current program mode as *ProgramMode*.

mode
The current mode as *Mode*.

switch_modes(*standard_mode*, *substitute_mode*)
Switch current *mode* between *standard_mode* and *substitute_mode* as *Mode*. If current *mode* is none of these, it will be set to *standard_mode*.

comp
The current component *sim_model* for element creation when *mode* is set to *Mode.comp*.

raster_mode
Raster visibility as `bool`.

comp_raster
Raster visibility for composition mode as `bool`.

Use this property to memorize *raster_mode* before simulation and set it back when switching back to composition mode.

raster_snap_mode
`bool` property whether elements snap to raster positions on `mouse_release()`.

force_dragging
`bool` property whether scenario items are currently beeing dragged by an layout algorithm.

selection_dragging
`bool` property whether scenario items are currently beeing dragged by mouse selection.

selection
A list of elem_ids of selected elements.

is_selectable(*elem_or_elem_id*)
Return whether element is selectable according to current *mode*.

init_history()
Reset all history entries and set an initial history entry as first scenario state.

add_history_point()
Adds a new history point if there have been some changes and clears redo history list.

undo()
Undo latest change.

redo()
Redo latest undone change.

changes_count()
Return an int value representing the current number of model state changes. This function is used for detecting history specific differences to the last temporary stored scenario.

History specific changes consider elements, positions, dockings, parameters and scenario simulation settings. Selection changes are not detected.

scenario
Get or apply a JSON compatible scenario dict with the following keys:

- *changes_count()*: number of history specific scenario changes.
- *uid*: id counters for element creation.
- *sim_start*: simulation start time **as string** (e.g. `'2014-10-20T00:00:00'`).

- *sim_end*: simulation end time **as string** (e.g. `'2014-10-24T23:59:59'`).
- *sim_step_size*: the current simulation step size in seconds (e.g. `1800` = 30 minutes).
- *elements*: the elements dict of elem_id to element instance.
- *selection*: the currently selected elements as dict from elem_id to element.

copy()
Return a new flat copied model including ...

- scenario
- components and simulators descriptions
- no connected events.

copy_to_clipboard(*elem_ids*)
Copy all elements of the elem_ids list to an internal clipboard.

paste_from_clipboard()
Paste all elements from internal clipboard and return the newly inserted elem_ids. Only dockings inside of clipboard will be maintained.

create_element(*sim_model*, *pos*)
Create a new element with a specific *sim_model* on an specific *position* as `QtCore.QPointF` in scenario.

delete_element(*elem_id*)
Delete the given element.

get_component(*elem_id*)
Return the component description dict of an *elem_id* by the elements *sim_model*.

get_simulator(*elem_id*)
Return the simulator description dict of an element.

get_icon_color(*elem_id*)
Return the color of the element icon. The icon will be scanned for the most valuable color only once. The icon colors are memorized locally.

get_shared_published_params(*elem_ids*)
Return a filtered list of published parameter names which are contained in each element of *elem_ids*.

param_is_multivalue(*elem_ids*, *param_name*)
Return whether one specific parameter (*param_name*) of multiple elements (*elem_ids*) differ in their values.

get_param_value(*elem_id*, *param_name*)
Get the value of a parameter in element (*elem_id*), Return `None` if element has no parameter *param_name*.

set_param_value(*elem_id*, *param_name*, *param_value*)
Set value of parameter (*param_name*) in element (*elem_id*) if value is not `None`.

get_shared_published_attrs(*elem_ids*)
Return a filtered list of published attribute names which are contained in each element of elem_ids.

attr_is_multivalue(*elem_ids*, *attr_name*)
Return whether current attribute values differ in elements of *elem_ids*.

get_attr_values(*elem_id*, *attr_name*, *from_time_index=0*, *to_time_index=None*)
Get attribute values of *attr_name* in element (*elem_id*) in time interval [*from_time_index*, *to_time_index*] where *to_time_index* is *sim_timestamp* if set to `None`.

get_attr_value(*elem_id*, *attr_name*, *time_index=None*)
Get current attribute value of *attr_name* in element (*elem_id*) at current time index. Return `None` if attribute does not exist.

get_u_heat_value(*elem_id*)
Return the **voltage** dependent heat value of an element with component type PQBus or Branch.

get_i_heat_value(*elem_id*)
Return the **current** dependent heat value of an element only with component type Branch.

get_p_level(*elem_id*)
Return the **power** level of an element with component type House, PV, Transformer, CHP, WECS or EV.

get_state_of_charge(*elem_id*)
Return the state of charge of an EV.

get_selected(*elem_id*)
Return whether element (*elem_id*) is selected.

set_selected(*elem_id*, *value*)
Set whether element (*elem_id*) is selected.

docking_port(*ep*)
Return an element docking port content.

Example `docking_port(['CSV.House_1', '1'])` returns the port '1' content of the element docking ports:

```
{
    'pos': [220.0, 330.0],
    'in': [],
    'out': [['PyPower.PQBus_3', '0']]
}
```

elem_ports(*elem_id*)
Return a list of available element ports of an specific element (*elem_id*).

get_pos(*ep*)
Get the position of an element port as `QtCore.QPointF`.

set_pos(*ep*, *pos*)
Set a position of an element port as `QtCore.QPointF`.

dockings_out(*ep*)
Return a list of outgoing dockings to other element ports from an element port *ep*.

dockings_in(*ep*)
Return a list of ingoing dockings from other element ports to an element port *ep*.

docking_attrs(*from_elem_id*, *to_elem_id*)
Return a set of valid attribute connection tuples from one element to another, e.g. `{('P_out','P')}`. This function is used for connecting elements in mosaik in *maverig.models.modelSimulation.SimulationProcess.start_simulation()*.

Component attribute descriptions may contain `'out':[...]` or `'in':[...]` entries indicating allowed connections to or from other attribute names.

can_dock(*from_ep*, *to_ep*)
Check whether it is possible to dock from one port to another.

dock(*from_ep*, *to_ep*)
Dock one port to another port.

undock (*from_ep*, *to_ep*)
Undock two element ports. Only undock if elements exists, because undock might have been called by element deletion.

handle_scenario_error (*e*)
Creates output and error events and selects elements in elem_ids for visual feedback of scenario errors.

validate_scenario ()
Validates the scenario.

update ()
Fires all events with pending demands.

update_all ()
Fires all events.

deselect_all_elems ()
Deselect all selected elements.

stop_simulation ()
Stop the simulation and switch the mode to selection and program_mode to composition.

maverig.models.modelGraph

class maverig.models.modelGraph.**ModelGraph** (*model*, *data=None*, ***attr*)
Bases: networkx.Graph

A networkx.Graph representation of the *model element ports* as *nodes* and element internal lines between them as *edges*.

This graph can be used for layout optimization algorithms.

Example connection scenario:

```
('House', '0') --- ('House', '1') -> ('PQBus_1', '0')

('PQBus_1', '0') <- ('Branch', '0') --- ('Branch', '1') -> ('PQBus_2', '0')
```

Where `---` are lines and `<-`, `->` are dockings.

This scenario would be represented in ModelGraph as follows:

```
('House', '0') --- ('PQBus_1', '0') --- ('PQBus_2', '0'))
```

Where `---` are edges and the element ports are nodes.

Ports with outgoing dockings (endpoints in view) get represented by their docked Port as Node, because they share the same position.

set_pos (*ep*, *pos*)
Change the position of an ElemPort node to pos (PySide.QtCore.QPointF). This method gets called by model when a position is set there.

maverig.models.modelSimulation

class maverig.models.modelSimulation.**SimulationProcess** (*sim_proxy*, *model*)
Bases: multiprocessing.context.Process

The Simulation Process instantiates all needed Mosaik models and simulators and runs them until simulation is finished.

run()
Process entry point.

short_name(*name*)
Strip name extensions after '-'.

start_simulation()
Start simulators and their elements in Mosaik and create and run Mosaik world.

The Simulation is beeing initiated in the following order:

1. Prepare model and simulator specific parameters and configuration.
2. Start the simulators with parameter values taken from element where possible.
3. Start the element as Mosaik entity and optionally create needed parent entities
4. Connect entities as defined by dockings and component attribute descriptions
5. Create and connect *maverig.utils.visSimulator.VisSimulator* which constantly sends simulated data to *SimulationServer* on each simulation step.
6. Run the Simulation and return when it is finished.

start_simulator(*world*, *elem*)
Starts a *mosaik_simulator* with parameters specified in prepared element simulator parameter dict (`elem['simulator']['params']`).

start_element(*elem*)
Create an element in Mosaik with needed params specified in simulator meta and collected from *elem*. Apply the mapping of mosaik element id (*mosaik_full_id*) to *elem*, so that the data pushed to *SimulationServer* by *maverig.utils.visSimulator.VisSimulator* can be mapped back to *elem_id* on *SimulationServer.run_iteration()* side later on.

class maverig.models.modelSimulation.**SimulationServer**(*model*)
Bases: *maverig.utils.processServer.ProcessServer*

Manage a simulation process and serve proxy-access to registered functions.

start()
Reset attributes and start class:*SimulationProcess* as Maverig subprocess.

stop()
Stop the Simulation and terminate the simulation process if it is still active.

write(*text*)
Proxy function. Write text and subprocess stdout to Maverig console output.

handle_process_scenario_error(*e*)
Proxy function for error handling passed to *maverig.models.model.Model.handle_scenario_error()*.

map_elem_to_mosaik(*elem_id*, *mosaik_full_id*)
Proxy function mapping *mosaik_full_id* to *elem_id* for data assignment in *run_iteration()*.

update_data(*timestamp*, *progress*, *data*)
Proxy function for pushing collected from *maverig.utils.visSimulator.VisSimulator*. Apply the data later in *run_iteration()* by starting a timer. This may be faster than applying the data on each processed *update_data* call.

run_iteration()
Apply simulated data on model.

maverig.presenter

maverig.presenter.group_presenter

- maverig.presenter.group_presenter.abstractGroupPresenter
- maverig.presenter.group_presenter.iconGroupPresenter
- maverig.presenter.group_presenter.lineGroupPresenter
- maverig.presenter.group_presenter.lineIconGroupPresenter
- maverig.presenter.group_presenter.nodeGroupPresenter

maverig.presenter.group_presenter.abstractGroupPresenter

class maverig.presenter.group_presenter.abstractGroupPresenter.**AbstractGroupPresenter**(*presenter_model*, *elem_id*, *cfg*)

Bases: maverig.presenter.abstractPresenter.AbstractPresenter

Presenter class that acts as the event handler between the view and the model for all groups.

mappings_port_vp
Maps the port and the v_point.

ep(*v_point*)
Returns the element port belonging to a virtual point.

vp(*elem_port*)
Returns the virtual point belonging to an element port.

raster_snap_v_points
Returns a list of virtual points that may snap to the raster.

init_scene_mapping(*scene*)
Adds view items to the scene, sets related elem_id as tooltip for easier handling of errors and triggers scene mapping.

remove()
Removes this whole group. Unsubscribes model events.

snap_zone(*v_point*, *pos=None*)
Returns a list of nearby virtual points (of other groups) sorted by distance (from near to far).

can_dock(*from_vp*, *to_vp*)
Returns if virtual point from_vp can dock with virtual point to_vp if docking is accepted and from_vp is not docked to another virtual point already. A virtual point (e.g. endpoint) can have one outgoing connection only but may have several ingoing connections (e.g. node).

to_dockables(*from_vp*, *to_vps*)
Returns a list of virtual points from to_vps to which the virtual point from_vp can dock-out.

from_dockables(*from_vps*, *to_vp*)
Returns a list of virtual points from from_vps to which the virtual point to_vp can dock-in.

connectables(*other_vps*, *vp*)
Returns a list of virtual points from other_vps to which the virtual point vp can dock-in or dock-out.

new_connectables(*other_vps*, *vp*)
Returns a list of virtual points from other_vps to which the virtual point vp can dock-in or dock-out if they are not docked already.

non_connectable(*other_vps*, *vp*)
Returns a list of virtual points from other_vps to which the virtual point vp can't dock.

dock(*from_vp*, *to_vp*)
Docks virtual points and applies docking in model.

undock(*from_vp*, *to_vp*)
Undocks virtual points and applies undocking in model.

check_snap_permission(*do_validate=False*)
Checks whether the view has permission to snap. Optionally validates snap restrictions.

on_position_changed(*vp*, *delta*, *change*, *section*)
Applies docking/undocking of the view within snap zone. Sets v_point position in model.

on_exit_presenter_section(*section*)
Informs the user about docking validity via status bar messages.

validation(*vp*, *snap_zone=None*)
Updates the views validity.

snap_dock()
Snaps the view to the raster.

raster_snap()
Snaps the view to the raster if raster mode is enabled.

avoid_invalid_positions()
Avoids invalid positions by moving the view back to last valid positions.

on_mouse_released(*mouse_pos*)
Applies raster snapping to the view if view is released by the mouse.

on_elements()
Reacts on changes on elements count and removes himself if view isn't present anymore.

on_positions()
Reacts on position changes.

on_drag()
Reacts on drag and drop.

on_selection()
Reacts on selection changes. Updates the views z-mode and visibility state.

on_dockings()
Reacts on view docking.

on_mode()
Reacts on mode changes. Updates the state of the view.

on_param()
Reacts on parameter changes.

change_ev_icon(*value*)
Changes the icon of an electric vehicle depending on state of charge and plugged-in state.

set_effect(*view_or_item*, *effect*, *color=<PySide.QtGui.QColor object>*, *value=0*)
Applies heat value effects to the view. :param color: the color- or shadow-effect color :param value: value for bar width or opacity (0..1)

on_attrs()
Reacts on value changes of attributes. Triggers applying of heat value effects to the view.

maverig.presenter.group_presenter.iconGroupPresenter

class maverig.presenter.group_presenter.iconGroupPresenter.**IconGroupPresenter**(*presenter_manager*, *model*, *elem_id*, *cfg*)
Bases: *maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter*

Presenter class that acts as the event handler between the view and the model for the icon group.

mappings_port_vp
Maps the port and the v_point, e.g. {'0': icon.vp_center}.

maverig.presenter.group_presenter.lineGroupPresenter

class maverig.presenter.group_presenter.lineGroupPresenter.**LineGroupPresenter**(*presenter_manager*, *model*, *elem_id*, *cfg*)
Bases: *maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter*

Presenter class that acts as the event handler between the view and the model for the line and line icon group.

mappings_port_vp
Maps the port and the v_point, e.g. {'0': endpoint_left.vp_center, '1': endpoint_right.vp_center}.

on_param()

maverig.presenter.group_presenter.lineIconGroupPresenter

maverig.presenter.group_presenter.nodeGroupPresenter

class maverig.presenter.group_presenter.nodeGroupPresenter.**NodeGroupPresenter**(*presenter_manager*, *model*, *elem_id*, *cfg*)
Bases: *maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter*

Presenter class that acts as the event handler between the view and the model for the node group.

mappings_port_vp
Maps the port and the v_point, e.g. {'0': node.vp_center}.

maverig.presenter.utils

- *maverig.presenter.utils.forceEngine*

maverig.presenter.utils.forceEngine

class maverig.presenter.utils.forceEngine.**ForceEngine**(*scenario_panel_presenter*, *model*, *scene*)

Bases: object

run_iteration()

apply_positions()
applies node position movements on mapped virtual points in scene

- *maverig.presenter.abstractPresenter*
- *maverig.presenter.attributePanelPresenter*
- *maverig.presenter.modePanelPresenter*
- *maverig.presenter.componentWizardPresenter*
- *maverig.presenter.consolePanelPresenter*
- *maverig.presenter.menuBarPresenter*
- *maverig.presenter.presenterManager*
- *maverig.presenter.progressPresenter*
- *maverig.presenter.propertyPanelPresenter*
- *maverig.presenter.scenarioPanelPresenter*
- *maverig.presenter.settingsPresenter*
- *maverig.presenter.statusBarPresenter*
- *maverig.presenter.toolbarPresenter*

maverig.presenter.abstractPresenter

class maverig.presenter.abstractPresenter.**AbstractPresenter**(*presenter_manager*, *model*, *cfg=None*)

Bases: object

on_settings()
Reacts on settings respectively config changes.

maverig.presenter.attributePanelPresenter

class maverig.presenter.attributePanelPresenter.**AttributePanelPresenter**(*presenter_manager*, *model*, *cfg*)

Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the attribute panel.

on_change_visibility_triggered()
Toggles the visibility of the attribute panel. Saves the visibility state in the config.

on_change_graph_visibility_triggered(*name*)
Toggles the visibility of the different graphs in the attribute panel.

update_graph_visibility(*name*)
Saves the visible graphs in the config.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_selection()
Reacts on selection changes. Creates attribute cells for the attribute panel.

on_attrs()
Reacts on value changes of attributes. Updates the displayed values in the attribute panel.

on_program_mode()
Reacts on program mode changes. In the composition program mode the attribute panel is hided while the panel is visible in the simulation program mode if the user didn't hide it.

maverig.presenter.modePanelPresenter

class maverig.presenter.modePanelPresenter.**ModePanelPresenter**(*presenter_manager*, *model*, *cfg*)
Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the mode panel.

on_btn_context_menu(*event*, *comp_name*)
Creates and opens a context menu when the user performs a right mouse click on a component button.

on_context_menu(*event*)
Creates and opens a context menu when the user performs a right mouse click in the component panel.

remove_selected_component()
Removes a component.

hide_selected_component()
Hides a component.

show_invisible_components()
Toggles the visibility of hided components in component panel.

restore_default_components()
Removes all components from and restores the default components. The list of hided components in the config gets cleared.

selection_mode_btn_clicked()
Switches the mode between 'selection mode' and 'component mode' when the selection mode button is clicked.

hand_mode_btn_clicked()
Switches the mode between 'hand mode' and 'component mode' when the hand mode button is clicked.

add_component_btn_clicked()
Opens the component wizard.

comp_btn_created(*btn*, *comp_name*)
Adds created button to buttons dict.

comp_btn_clicked(*btn*, *comp_name*)
Switches the mode between 'component mode' and 'selection mode' when a component button is clicked.

drag_started(*btn*, *comp_name*)
Switches the mode to component mode when a component is dragged.

on_change_visibility_triggered()
Toggles the visibility of the component panel. Saves the visibility state in the config.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_mode()
Reacts on mode changes. Updates the component buttons accordingly.

on_program_mode()
Reacts on program mode changes. In the simulation program mode the component panel is hided while the panel is displayed in the composition program mode if the user didn't hide it.

on_components()
Reacts on component changes. Triggers view reinitialization so that the panel adopts the changes.

get_published_components()
Returns a list with comp_name, category, icon and tooltip of every existing component for component grid creation.

maverig.presenter.componentWizardPresenter

class maverig.presenter.componentWizardPresenter.**ComponentWizardPresenter**(*presenter_manager*, *model*, *cfg*)

Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the component wizard.

get_simulator_names()
Return the names of available simulators.

get_category_names()
Return a sorted list with the names of the categories.

on_new_simulator_triggered()
Triggers the visibility of the attribute panel.

init_view(*v*)

on_add_simulator_triggered()
Save new simulator description in json.

There are some TODOs in this method as advice future feature development.

on_add_component()
Save new component description in json.

There are some TODOs in this method as advice future feature development.

on_close_wizard()
Whenever the dialog gets closed (cancel, finish or close event), switch back to last known mode.

maverig.presenter.consolePanelPresenter

class maverig.presenter.consolePanelPresenter.**ConsolePanelPresenter**(*presenter_manager*, *model*, *cfg*)

Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the console panel.

on_change_visibility_triggered()
Toggles the visibility of the console panel. Saves the visibility state in the config.

on_console_clear_triggered()
Clears the console output.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_program_mode()
Reacts on program mode changes. The console panel is visible in every program mode if the user didn't hide it.

on_output(*output*, *new_line=True*)
Appends the given output to the console output.

maverig.presenter.menuBarPresenter

class maverig.presenter.menuBarPresenter.**MenuBarPresenter**(*presenter_manager*, *model*, *cfg*)
Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the menu bar. Functionality that is concerning the scenario is realized in 'maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter'.

path = None

on_file_new_triggered()
Discards the current scenario.

new_file()
Clears the current scenario.

on_file_open_triggered()
Opens a file dialog.

open_file()
Loads a serialized scenario from a chosen file.

on_file_save_triggered()
Saves a serialized scenario. If the current scenario isn't saved within a file already a file dialog will be opened.

on_file_save_as_triggered()
Opens a file dialog so that the serialized scenario can be saved within a named file.

static on_quit_triggered()
Shuts down the application.

on_undo_triggered()
Undo latest change.

on_redo_triggered()
Redo latest undone change.

on_cut_triggered()
Cuts selected elements.

on_copy_triggered()
Copies selected elements.

on_paste_triggered()
Pastes copied elements and selects them.

on_delete_triggered()
Removes selected elements.

on_select_all_triggered()
Selects all elements depending on current mode.

on_back_to_start_triggered()
Sets progress slider position to first index.

on_reduce_speed_triggered()
Reduces the speed of the progress slider.

on_run_triggered()
Starts the simulation and runs or pauses the progress slider. Switches the program mode.

on_stop_triggered()
Stops the simulation and progress slider. Switches to the composition program mode.

on_pause_triggered()
Pauses the progress slider and switches to the simulation paused program mode.

on_increase_speed_triggered()
Increases the speed of the progress slider.

on_forward_to_end_triggered()
Sets progress slider position to last possible index.

on_set_time_triggered()
Opens a dialog for changing time and speed parameters of the simulation. Sets the start time, the end time, the step size and the progress slider speed returned from the dialog.

on_go_to_triggered()
Opens a dialog where the user can set the progress slider position to a specific simulation time.

on_hand_mode_triggered()
Toggles the hand mode for shifting the scenario.

on_selection_mode_triggered()
Toggles the selection mode for element selection.

on_raster_mode_triggered()
Toggles raster mode for element snapping and scenario raster.

on_zoom_in_triggered()
Scales up the scenario.

on_zoom_out_triggered()
Scales down the scenario.

on_zoom_fit_triggered()
Fits all elements into the view.

on_trigger_component_panel()
Toggles the visibility of the component panel.

on_trigger_property_panel()
Toggles the visibility of the property panel.

on_trigger_console()
Toggles the visibility of the console panel.

on_trigger_status_bar()
Toggles the visibility of the status bar.

on_trigger_progress_bar()
Toggles the visibility of the progress bar.

on_trigger_attribute_panel()
Toggles the visibility of the attribute panel.

on_auto_layout_triggered()
Triggers scenario redrawing with ForceAtlas2.

on_settings_triggered()
Opens the settings dialog.

static on_help_triggered()
Opens the User Manual.

static on_about_triggered()
Opens the about dialog.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_elements()
Reacts on changes of the elements count and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_mode()
Reacts on mode changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_drag()
Reacts if a component is dragged and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_vid_speed()
Reacts on changes of the progress slider speed and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_program_mode()
Reacts on program mode changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_selection()
Reacts on selection changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_clipboard()
Reacts on clipboard changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

static datetime_to_qdatetime(*date_time*)

static qdatetime_to_datetime(*qdatetime*)

maverig.presenter.presenterManager

class maverig.presenter.presenterManager.**PresenterManager**(*model*, *cfg*)
Bases: object

Container class for all presenters. Each presenter may call methods on other presenters through this container.

maverig.presenter.progressPresenter

class maverig.presenter.progressPresenter.**ProgressPresenter**(*presenter_manager*, *model*, *cfg*)

Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the progress bar.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_slider_moved(*position*)
Sets simulation data index to current slider position. Keeps slider position valid if the mosaik simulation progress isn't as far as the position. The model performs lazy updates on the UI through the refresh_timer to prevent application from speed and graph animation issues.

on_change_visibility_triggered()
Toggles the visibility of the progress bar. Saves the visibility state in the config.

on_change_dateformat()
Toggles displaying of the date.

on_screen_dateformat()
Reacts on changes of the date display. Displays the date as calendar date or as countdown.

on_progress()
Applies the current mosaik simulation progress to the progress bar.

on_sim()
Reacts on simulation data index changes. Updates the slider position and the date.

on_vid_speed()
Reacts on changes of the progress slider speed. Stops und starts the progress slider to adopt new speed.

on_program_mode()
Reacts on program mode changes. In the composition program mode the progress bar is hided while the progress bar is visible in the simulation program mode if the user didn't hide it. In addition the progress slider and the progress bar are set back if application switches to the composition program mode.

run_slider()
Starts the progress slider.

stop_slider()
Stops the progress slider.

run_iteration()
Updates the simulation data index which is responsible for moving the progress slider.

run_refresh()
Connected to timer which is responsible for model updates.

maverig.presenter.propertyPanelPresenter

class maverig.presenter.propertyPanelPresenter.**PropertyPanelPresenter**(*presenter_manager*, *model*, *cfg*)

Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the property panel.

on_change_visibility_triggered()
Toggles the visibility of the property panel. Saves the visibility state in the config.

check_spinbox(*widget*, *value*)
Checks spinbox values.

value_changed(*widget*, *value*)
Handles value changes of the properties.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_selection()
Reacts on selection changes. Triggers view reinitialization so that the view is updated depending on the selection.

on_param()
Reacts on value changes in model and updates the view.

on_program_mode()
Reacts on program mode changes. In the simulation program mode the property panel is hided while the panel is visible in the composition program mode if the user didn't hide it.

maverig.presenter.scenarioPanelPresenter

class maverig.presenter.scenarioPanelPresenter.**ScenarioPanelPresenter**(*presenter_manager*, *model*, *cfg*)

Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the scenario panel.

adjust_scene_rect()
Fits the size of the scene to the elements bounding rect.

on_selection_changed()
Adopts selection changes in model. Adapts visibility and z-mode of elements.

on_context_menu(*event*)
Creates and opens a context menu when the user performs a right mouse click in the scenario panel.

on_draw_background(*painter*)
Triggers raster drawing if raster mode is enabled.

element_at(*mouse_pos*)
Returns element at the current mouse position.

create_new_element(*mouse_pos*)
Creates a new element at the given mouse position.

mouse_clicked(*mouse_pos*, *button*)
Triggers creation of a new element at the given mouse position by clicking left mouse button. Switches between selection mode and component mode by clicking right mouse button.

damped_mouse_pos(*mouse_pos*)
Returns the damped mouse position if it is out of frame rect.

mouse_moved(*mouse_pos*, *buttons*)
Sets mouse position based on the damped mouse position.

mouse_released(*mouse_pos*)
Draws second endpoint of a line.

zoom(*zoom_in*, *wheel_event=None*)
Scales up or scales down the scenario depending on the mouse wheel alpha.

zoom_fit()
Fits all elements into the view.

delete_selected_elements()
Removes all selected elements.

copy_selected_elements()
Copies all selected elements.

cut_selected_elements()
Cuts all selected elements.

paste_elements()
Pastes all copied elements and selects them.

select_all_elements()
Selects all elements.

select_all_active_elements(*mouse_pos*)
Selects all elements depending on the current active mode.

on_attrs()
Reacts on value changes of attributes. Updates the background visualization of the date time if day and night visualization is enabled.

on_mode()
Reacts on mode changes. Updates the views drag mode, interactive mode and mouse cursor depending on the active mode.

on_drag()
Reacts if an element is dragged into the scene from the component panel.

on_elements()
Reacts on changes on elements count and updates the view.

on_error(*title*, *text*, *info_text*, *elem_ids*)
Reacts on model scenario errors and displays an error dialog

run_force_layout()
Triggers running of the force atlas algorithm.

group_presenters
Returns a set of all group presenters.

groups
Returns a set of all groups.

maverig.presenter.settingsPresenter

class maverig.presenter.settingsPresenter.**SettingsPresenter**(*presenter_manager*, *model*, *cfg*)
Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the settings dialog.

on_change_visibility_triggered()
Shows the settings dialog.

on_language_changed()
Sets flag 'do_change_language' to true if the language has been changed. The flag is used when the presenter applies the settings.

install_language(*do_update=True*)
Sets chosen language if it is changed by user. This is handled separately to prevent handling of unnecessary events in whole application if the language hasn't been changed.

apply_settings()
Triggers applying of settings.

apply_setting(*tab*, *setting*, *do_update=True*)
Applies the given setting.

maverig.presenter.statusBarPresenter

class maverig.presenter.statusBarPresenter.**StatusBarPresenter**(*presenter_manager*, *model*, *cfg*)
Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for the status bar.

on_change_visibility_triggered()
Toggles the visibility of the status bar. Saves the visibility state in the config.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_drag()
Reacts if an element is dragged. Displays information that dragging is active currently.

on_mode()
Reacts on mode changes. Displays the chosen mode in the status bar.

on_vid_speed_event()
Reacts on changes of the progress slider speed. Displays the current speed in the status bar.

on_program_mode()
Reacts on program mode changes. The status bar is visible in every program mode if the user didn't hide it.

error(*message*)
Sets the given message in the status bar and applies a red background to the status bar.

info(*message*)
Sets the given message in the status bar and applies a blue background to the status bar..

success(*message*)
Sets the given message in the status bar and applies a green background to the status bar..

reset()
Resets the status bar.

maverig.presenter.toolbarPresenter

class maverig.presenter.toolbarPresenter.**ToolbarPresenter**(*presenter_manager*, *model*, *cfg*)
Bases: *maverig.presenter.abstractPresenter.AbstractPresenter*

Presenter class that acts as the event handler between the view and the model for toolbar. Events that are fired by the toolbar view are mapped to the menu bar presenter because all functionality of the toolbar is covered by the menu bar. Code changes can be realized at one place in this way.

on_file_open_triggered()
Opens a file dialog.

on_file_save_triggered()
Saves a serialized scenario. If the current scenario isn't saved within a file already a file dialog will be opened.

on_back_to_start_triggered()
Sets progress slider position to first index.

on_reduce_speed_triggered()
Reduces the speed of the progress slider.

on_run_triggered()
Starts the simulation and runs or pauses the progress slider. Switches the program mode.

on_stop_triggered()
Stops the simulation and progress slider. Switches to the composition program mode.

on_increase_speed_triggered()
Increases the speed of the progress slider.

on_forward_to_end_triggered()
Sets progress slider position to last possible index.

on_zoom_in_triggered()
Scales up the scenario.

on_zoom_out_triggered()
Scales down the scenario.

on_zoom_fit_triggered()
Fits all elements into the view.

on_delete_triggered()
Removes selected elements.

on_settings_triggered()
Opens the settings dialog.

on_auto_layout_triggered()
Triggers scenario redrawing with ForceAtlas2.

on_language()
Reacts on language changes. Triggers view reinitialization so that the view adopts the chosen language.

on_elements()
Reacts on changes of the elements count and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_selection()
Reacts on selection changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_drag()
Reacts if a component is dragged and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_sim()
Reacts on changes of the simulation time and speed parameters and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_vid_speed()
Reacts on changes of the progress slider speed and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

on_program_mode()
Reacts on program mode changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

maverig.views

maverig.views.groups

- *maverig.views.groups.abstractGroup*
- *maverig.views.groups.iconGroup*
- *maverig.views.groups.lineGroup*
- *maverig.views.groups.lineIconGroup*
- *maverig.views.groups.nodeGroup*

maverig.views.groups.abstractGroup

class maverig.views.groups.abstractGroup.**AbstractGroup**
Bases: *maverig.views.abstractView.AbstractView*

init_view(*scene=None*)

add_to_scene(*scene*)
add subitems to scene

add_item(*item*)

remove_item(*item*)

selected

enabled

clear_effects()

clear_state_of_charge_effect()

clear_state_of_charge_tip()

clear_state_of_charge_tip_bg()

set_color_effect(*color*, *transparency*)

set_shadow_effect(*color*, *shadow_faint*, *offset1*, *offset2*)

set_consumer_bar_effect(*color*, *pos*, *width*, *height*)

set_producer_bar_effect(*color*, *pos*, *width*, *height*)

set_state_of_charge_bar(*color*, *pos*, *width*, *height*)

set_state_of_charge_tip(*color*, *pos*, *width*, *height*)

set_state_of_charge_tip_bg(*color*, *pos*, *width*, *height*)

is_under_mouse

remove()

maverig.views.groups.iconGroup

class maverig.views.groups.iconGroup.**IconGroup**(*positions*, *icon_path*)
Bases: *maverig.views.groups.abstractGroup.AbstractGroup*

add_endpoint(*endpoint_pos*)
add an endpoint with a dotted line to icon

maverig.views.groups.lineGroup

class maverig.views.groups.lineGroup.**LineGroup**(*positions*)
Bases: *maverig.views.groups.abstractGroup.AbstractGroup*

maverig.views.groups.lineIconGroup

class maverig.views.groups.lineIconGroup.**LineIconGroup**(*positions*, *icon_file*)
Bases: *maverig.views.groups.abstractGroup.AbstractGroup*

maverig.views.groups.nodeGroup

class maverig.views.groups.nodeGroup.**NodeGroup**(*positions*)
Bases: *maverig.views.groups.abstractGroup.AbstractGroup*

maverig.views.items

- *maverig.views.items.abstractItem*
- *maverig.views.items.circle*
- *maverig.views.items.icon*
- *maverig.views.items.line*

maverig.views.items.abstractItem

class maverig.views.items.abstractItem.**AbstractItem**(*parent_group*)
Bases: object

init_graphics_item()
post initialization of graphics_item. This method needs to be called after graphics_item initialization in subclasses.

add_to_scene(*scene*)

add_v_point(*v_point*)

z_value

visible

enabled

opacity

set_color_effect(*color*, *transparency*)

set_shadow_effect(*color*, *shadow_faint*, *offset1*, *offset2*)

clear_effects()

clear_state_of_charge_effect()

clear_state_of_charge_tip()

clear_state_of_charge_tip_bg()

set_consumer_bar_effect(*color*, *pos*, *width*, *height*)

set_producer_bar_effect(*color*, *pos*, *width*, *height*)

set_state_of_charge_bar(*color*, *pos*, *width*, *height*)

set_state_of_charge_tip(*color*, *pos*, *width*, *height*)

set_state_of_charge_tip_bg(*color*, *pos*, *width*, *height*)

selected

selectable

is_under_mouse

move_pos(*delta*)

on_position_changed(*vp*, *delta*)

remove()

maverig.views.items.circle

class maverig.views.items.circle.**Circle**(*parent_group*, *pos*, *style*)

Bases: *maverig.views.items.abstractItem.AbstractItem*

init_graphics_item()

move_pos(*delta*)

on_position_changed(*vp*, *delta*, *change*, *section*)

circle_style

maverig.views.items.icon

class maverig.views.items.icon.**Icon**(*parent_group*, *pos*, *icon_path*)

Bases: *maverig.views.items.abstractItem.AbstractItem*

init_graphics_item(*icon_path*)

icon_path

move_pos(*delta*)

on_position_changed(*vp*, *delta*, *change*, *section*)

maverig.views.items.line

class maverig.views.items.line.**Line**(*parent_group*, *start_pos*, *end_pos*, *line_style*)
Bases: *maverig.views.items.abstractItem.AbstractItem*

init_graphics_item()

move_pos(*delta*)

on_position_changed(*vp*, *delta*, *change*, *section*)

adjust_line()
adjusts the line position to the endpoints.

line_style

maverig.views.positioning

- *maverig.views.positioning.section*
- *maverig.views.positioning.vPoint*

maverig.views.positioning.section

class maverig.views.positioning.section.**Section**(*name*, *next_sequence=None*, *called_by_remote=False*)
Bases: object

synchronization of critical sections and automatic walk trough specified section sequences. All section participants are notified via events when a section is entered and leaved (exit).

Sections can either be started directly (run) or be initiated remotely by calling enter and exit.

run()

is_running()

enter()
first call enters section

exit(**args*)
last call exits section

class maverig.views.positioning.section.**SectionManager**
Bases: object

Holds a section sequence for positioning on internal, items and presenter side.

Sections divide position_changed-Event-Handling into multiple subjected Phases which will be handled one after another.

Position changes made during an `with section_manager.pos_section`-block will only throw events after all `pos_section`-blocks have been exited and the next section (`mouse_section`) has been entered automatically.

Then all changed virtual points will throw a position_changed_event in `__on_enter_section` with the actual section as parameter which can be checked on Event-Handling.

Afterwards all changed virtual points will throw a `position_changed_event` for the next section and so on...

This is, how positioning works with VPoints and SectionManager:

1. **pos_section:** in order to set multiple VPoint positions at the same time, use:

```
with section_manager.pos_section:
    v_point1.set_pos(pos1, Change.moved)
    v_point2.pos = pos2  # Change == Change.applied
```

This is equivalent to:

```
section_manager.pos_section.enter
v_point1.set_pos(pos1, Change.moved)
v_point2.pos = pos2  # Change == Change.applied
section_manager.pos_section.exit
```

one VPoint can simply be set like this:

```
v_point.pos = pos  # Change == Change.applied
```

In this case, the `pos_section` *enter* and *exit* will be called remotely by VPoint position setter.

During the time window marked by `[pos_section.enter .. adjust_section.exit]`, all position changes are applied on old position values in order to prevent side-effects.

2. **mouse_section:** Mouse section is entered directly after all positioning has been applied on pos_section. (`VPoint.__on_enter_section`)

   Any mouse moved virtual point triggers adjustment (as specified in `VPoint.trigger_section`) on mouse_section.

3. **adjust_section:** Adjustment section is entered after primary positioning and mouse moved positions.

   `VPoint` followings and fixings will be applied by handling `VPoint.position_changed` on previous position changes.

   `VPoint.pos` calls still return the old position. The new set position is saved internally at `VPoint.__new_pos` and will be applied to `VPoint.pos` after all adjustment changes have been finished. (`VPoint.__on_exit_section`)

4. **item_section:** Item section is entered after all affected positions and their followings/fixings have been updated. All item positions can now be adapted to VPoint positions by registering on `VPoint.position_changed` event:

```
def on_position_changed(self, vp, delta, change, section):
    if section == section_manager.item_section:
        self.graphics_item.setPos(self.graphics_item.pos() + delta)
```

5. **presenter_section:** Presenter section is entered after all affected item positions have been updated. GroupPresenter can now work on the current VPoint and QGraphicsItem positions and e.g. apply custom followings:

```
def on_position_changed(self, vp, delta, change, section):
    if section == section_manager.presenter_section:
        other_v_point = self.snap_zone_dockable(v_point)  # needs correct QGraphicsItem
        v_point.follow(other_v_point)
        other_v_point.follow(v_point, [Change.calculated, Change.raster_snapped])
```

The Presenter section may also apply new positions which restarts the section sequence from position level.

maverig.views.positioning.vPoint

class maverig.views.positioning.vPoint.**Changes**
Bases: object

none = {}

all = {'applied', 'moved', 'snapped', 'followed', 'calculated', 'avoid_invalid', 'raster_snapped'}

indirect = {'moved', 'snapped', 'followed', 'calculated', 'avoid_invalid', 'raster_snapped'}

class maverig.views.positioning.vPoint.**Change**
Bases: object

Reason for v_point change.

origin = None

applied = 'applied'

moved = 'moved'

snapped = 'snapped'

raster_snapped = 'raster_snapped'

avoid_invalid = 'avoid_invalid'

calculated = 'calculated'

followed = 'followed'

class maverig.views.positioning.vPoint.**VPoint**(*parent_item=None*)
Bases: object

A Virtual Point which represents a position in scenario.

Each Virtual Point can follow other Virtual Points via event triggers and adjustment functions on specific changes.

When setting a position, Position Changed Events are fired section by section through stacked layers, where each section stands for a different type of position adjustments that would conflict each other if they would all run in one section. The position gets applied on *pos* when the internal layer section_manager.adjust_section is being exited. This is in order to make multiple relative position changes based on the previous position without summing up the movements which would result in negative side effects.

See *maverig.views.positioning.section.SectionManager* documentation for how positioning works in detail.

position_changed = None
The position changed event with current delta (QtCore.QPointF), change (*Change*) and section (*maverig.views.positioning.section.Section*) as params.

last_positions = None
A list of last valid positions controlled by *maverig.presenter.group_presenter.abstractGroupPresente*
in order to undo position changes to not allowed positions.

trigger_section = None
The section layer (*maverig.views.positioning.section.Section*) on which adjustment to other virtual points movements should be done.

new_pos = None
The last proposed new position.

delta = None
The last proposed position change distance.

change = None
The last proposed *Change* reason of position change.

parent_item = None
The parent *maverig.views.items.abstractItem.AbstractItem* item where this virtual point is added to `AbstractItem.v_points`.

followers = None
Virtual points that follow this VPoint. A dict of virtual point to adjustment method that reacts on *position_changed* events.

pos
The position of the virtual point.

Use *set_pos()* if you want to change the position with a specified *change* reason. Setting this property will result in a simple *Change.applied* *change* reason.

set_pos(*value*, *change*)
Set position (`QtCore.QPointF`) with *change* reason.

move_pos(*delta*, *change*)
Move by *delta* (`QtCore.QPointF`) with *change* reason.

follow(*v_point*, *trigger_changes={'moved', 'snapped', 'followed', 'calculated', 'avoid_invalid', 'raster_snapped'}*, *result_change='followed'*, *keep_distance=True*)
Follow *v_point*.

Adjust this VPoint when *v_point* position change applies to *trigger_changes*. This VPoint will change it's position with the given *result_change* reason.

Set *keep_distance* to `False` if VPoint should only be moved relatively to v_point movements. Otherwise distance to v_point is being fixed as `QtCore.QPointF`-vector from now on.

unfollow(*v_point*)
Stop following *v_point*.

follows(*v_point*)
Return whether this VPoint follows *v_point* when *v_point* moves.

fix(*v_point*)
Follow v_point and vice versa.

unfix(*v_point*)
Loose any attachments to *v_point* and vice versa.

extern_followers
Return a list of virtual points of other groups that follow this VPoint.

class `maverig.views.positioning.vPoint.`**VPMouse**
Bases: *maverig.views.positioning.vPoint.VPoint*

Virtual point representing the mouse position. All followers trigger on mouse_section before other following-adjustments, which will be done in adjust_section

- maverig.views.abstractView
- maverig.views.attributePanelView
- maverig.views.modePanelView
- maverig.views.componentWizardView
- maverig.views.consolePanelView
- maverig.views.dialogs
- maverig.views.mainWindow
- maverig.views.menuBarView
- maverig.views.progressView
- maverig.views.propertyPanelView
- maverig.views.scenarioPanelView
- maverig.views.settingsView
- maverig.views.statusBarView
- maverig.views.toolbarView

maverig.views.abstractView

class maverig.views.abstractView.**AbstractView**
Bases: object

maverig.views.attributePanelView

class maverig.views.attributePanelView.**AttributePanelView**
Bases: PySide.QtGui.QScrollArea, *maverig.views.abstractView.AbstractView*

Represents the attribute panel.

init_ui()
Init the ui structure for the attribute panel.

create_attribute_panel()
Initially create the container layout of the attribute panel.

create_attribute_cell(*name*, *caption*, *unit*, *step_size*, *lines_labels*, *lines_values*, *lines_colors*, *graph_available*)
Create an attribute cell for every attribute of a selected element.

clear_container()

update_info_label(*value*)

translate()

class maverig.views.attributePanelView.**AttributeCell**(*name*, *caption*, *unit*, *sim_step_size*, *lines_labels*, *lines_values*, *lines_colors*, *graph_available*, *view*)
Bases: PySide.QtGui.QGroupBox, *maverig.views.abstractView.AbstractView*

Represents one cell which is stored in the container for one attribute.

translate()

start_scoll_to_center_anim()

set_graph_visibility(*value*)

create_graph()

resizeEvent(*event*)
adjust graph labels after resize with single shot timer

on_draw_graph()
auto-adjust padding to lables Taken from http://matplotlib.org/faq/howto_faq.html#automatically-make-room-for-tick-labels. Workaround for pyplot.tight_layout() due to unpredictable ValueErrors and slowness issues. return True if graph has been redrawed on canvas.

draw_canvas()
draw matplotlib graph on canvas. return True if drawing has been successful

change_content(*current_value*, *multivalue*, *lines_values*)
Change and update the dynamic content of the attribute cell without recreating the ui.

maverig.views.modePanelView

class maverig.views.modePanelView.**ModePanelView**
Bases: PySide.QtGui.QScrollArea, *maverig.views.abstractView.AbstractView*

Represents the mode panel. A component can be added by activating the necessary comp mode in the scenarioPanel. A comp mode can be activated by clicking on the appropriate button.

init_ui()

create_components_grid(*published_components*)
return a grid layout with components according to component configuration

create_category_layout(*category*)

uncheck_buttons()
uncheck all buttons

create_button(*icon_path*, *tooltip*, *transparency=False*, *btn_w=55*, *btn_h=55*, *icn_w=35*, *icn_h=35*)

Parameters

- **icon_path** – complete icon path
- **tool_tip** – description of the button
- **btn_w** – button width
- **btn_h** – button high
- **icn_w** – icon widht
- **icn_h** – icon high

Returns qpushbutton

button_mouse_move(*btn*, *comp_name*, *mouse_event*)

button_mouse_pressed(*btn*, *comp_name*, *mouse_event*)

static set_svg_icon(*btn*, *icon_path*, *width*, *height*)
draws an svg icon on button btn

hover_component_button(*btn*, *icn_w=55*, *icn_h=55*)
if a component button has been selected it will increase its size

unhover_component_button(*btn*, *icn_w=35*, *icn_h=35*)
if a component button has been deselected it will get a smaller size

contextMenuEvent(*event*)

create_btn_context_menu()

create_context_menu()

maverig.views.componentWizardView

maverig.views.componentWizardView.**is_acceptable**(*l_edit*)

class maverig.views.componentWizardView.**IntroPage**(*pres*, *parent=None*)
Bases: PySide.QtGui.QWizardPage

initializes the first dialog to add general component information

icon_update = <MagicMock id='139681737534880'>

add_new_icon()

simulator_update()

category_update()

new_category_dialog()
Open a dialog to enter the new for a new component category.

check_state(**args*, ***kwargs*)

isComplete()

class maverig.views.componentWizardView.**AttributeParameterPage**(*pres*, *parent=None*)
Bases: PySide.QtGui.QWizardPage

add_new_parameter_tab()
If the button in the corner of the tab container is pushed there has to be a new tab for an individual parameter

add_new_attribute_tab()
If the button in the corner of the tab container is pushed there has to be a new tab for an individual attribute

isComplete()

class maverig.views.componentWizardView.**ConclusionPage**(*parent=None*)
Bases: PySide.QtGui.QWizardPage

initializePage()

class maverig.views.componentWizardView.**Tabs**(*tab_factory_method*)
Bases: PySide.QtGui.QTabWidget

tabInserted(*index*)

tabRemoved(*index*)

class maverig.views.componentWizardView.**ParameterTab**
Bases: PySide.QtGui.QWidget

state_checked = <PySide.QtCore.Signal object>

check_state()

is_ok()

class maverig.views.componentWizardView.**AttributeTab**
Bases: PySide.QtGui.QWidget

state_checked = <PySide.QtCore.Signal object>

check_state()

is_ok()

class maverig.views.componentWizardView.**ComboUpdatingBox**
Bases: PySide.QtGui.QComboBox

A Combo Box which provides a signal to infornm when the popup is shown.

popup_triggered = <PySide.QtCore.Signal object>

showPopup()
Overrides the shopPopup method to provide update functionality

maverig.views.consolePanelView

class maverig.views.consolePanelView.**ConsolePanelView**
Bases: PySide.QtGui.QScrollArea, *maverig.views.abstractView.AbstractView*

Represents console output. Every action triggered by the user is documented in the console output. This helps the user to follow his triggered actions.

init_ui()

translate()

maverig.views.dialogs

class maverig.views.dialogs.**SimulationTimeDialog**
Bases: object

Represents the simulation time dialog.

show(*sim_start*, *sim_end*, *sim_step_size*, *vid_speed*)

convert_slider_value(*val*)

maverig.views.dialogs.**go_to_time_dialog**(*sim_time_instances*, *sim_index*)
Shows a dialog to go to an specific simulation time

maverig.views.dialogs.**inform_dialog**()

maverig.views.dialogs.**error_dialog**(*title*, *text*, *info_text*)

maverig.views.dialogs.**about_dialog**()

maverig.views.dialogs.**participant**()

maverig.views.dialogs.**show_license**()

maverig.views.dialogs.**element_already_exist**(*string*)

maverig.views.mainWindow

class maverig.views.mainWindow.**MainWindow**(*cfg*)
Bases: PySide.QtGui.QMainWindow

Represents the starting point of the application. All view layouts within 'maverig/views' are linked inside this class.

closeEvent(*event*)

maverig.views.menuBarView

class maverig.views.menuBarView.**MenuBarView**
Bases: PySide.QtGui.QMenuBar, *maverig.views.abstractView.AbstractView*

Represents the menu bar.

init_ui()

maverig.views.progressView

class maverig.views.progressView.**ProgressView**
Bases: PySide.QtGui.QWidget, *maverig.views.abstractView.AbstractView*

Represents the progress bar.

init_ui()

translate()

maverig.views.propertyPanelView

class maverig.views.propertyPanelView.**PropertyPanelView**
Bases: PySide.QtGui.QScrollArea, *maverig.views.abstractView.AbstractView*

Represents the property panel. Every component has specific properties which are displayed within this panel. The user can change the properties of every single component within this panel.

init_ui()

create_property_label(*label*, *row*)

create_property_icon(*icon_path*, *row*)

delete_grid_widget(*index*)

clear_prop_grid()

create_integer_property_cell(*label*, *value*, *row*, *accepted_values*)

create_float_property_cell(*label*, *value*, *row*, *accepted_values*)

create_str_property_cell(*label*, *value*, *row*, *accepted_values*)

create_household_cell(*row*, *num_hh*)

change_household_cell(*row*, *num_hh*)

alter_integer_property_cell(*value*, *row*)

create_boolean_property_cell(*label*, *state*, *row*)

alter_boolean_property_cell(*row*, *state*)

create_file_property_cell(*label*, *value*, *row*)

alter_file_property_cell(*row*)

add_property_value_object(*l_edit*)

open_file_dialog(*l_edit*)

init_selection_counter(*count*)

change_color(*widget*, *color*)

set_new_accepted_value(*widget*, *new_value*)

set_parameter_style(*widget*, *multivalue*)
if multiple selected components have different values give a hint

property_grid

property_value_objects

static set_svg_icon(*btn*, *icon_path*, *width*, *height*)
draws an svg icon on button btn

maverig.views.scenarioPanelView

class maverig.views.scenarioPanelView.**ScenarioPanelView**
Bases: PySide.QtGui.QGraphicsView, *maverig.views.abstractView.AbstractView*

Represents the scenario panel.

init_ui()

drawBackground(*painter*, *rect*)

paint_datetime(*bgcolor_from*, *bgcolor_to*)

refreshBg()

draw_raster(*cell_size*, *painter*)
draws a raster depending on the chosen cell size on the given painter device

mouseDoubleClickEvent(*event*)

mousePressEvent(*event*)

mouseMoveEvent(*event*)
allow dragging newly created elements

mouseReleaseEvent(*event*)

wheelEvent(*wheel_event*)

dragEnterEvent(*event*)

dragMoveEvent(*event*)

dropEvent(*event*)

contextMenuEvent(*event*)

create_context_menu()

maverig.views.settingsView

class maverig.views.settingsView.**SettingsView**
Bases: *maverig.views.abstractView.AbstractView*

Represents the settings dialog.

show(*cfg*)
Shows a dialog to change the settings

maverig.views.statusBarView

class maverig.views.statusBarView.**StatusBarView**
Bases: PySide.QtGui.QScrollArea, *maverig.views.abstractView.AbstractView*

Represents the status bar.

init_ui()

maverig.views.toolbarView

class maverig.views.toolbarView.**ToolbarView**
Bases: PySide.QtGui.QToolBar, *maverig.views.abstractView.AbstractView*

Represents the toolbar.

init_ui()

maverig.utils

- *maverig.utils.colorTools*
- *maverig.utils.event*
- *maverig.utils.flowlayout*
- *maverig.utils.forceatlas2*
- *maverig.utils.logger*
- *maverig.utils.maverig_csv*
- *maverig.utils.numTools*
- *maverig.utils.processServer*
- *maverig.utils.scenarioErrors*
- *maverig.utils.tableWidgets*
- *maverig.utils.visSimulator*

maverig.utils.colorTools

maverig.utils.colorTools.**lab_color**(*color*)
Convert QColor, tuple, list or numpy-array to LabColor.

maverig.utils.colorTools.**q_color**(*color*)
Convert tuple, list or numpy-array and QGlobalColor to QColor.

maverig.utils.colorTools.**np_color**(*color*)
Convert tuple, list, QGlobalColor and QColor to numpy array.

maverig.utils.colorTools.**color_interp**(*x*, *values*, *colors*)
Return linear color interpolation of value x within values mapped to colors.

Example:

```
    x = 0.09
    values = [0.05, 0.08, 0.1]
    colors = [Qt.black, Qt.yellow, Qt.red]
    return QColor(255, 127, 0) # orange
```

maverig.utils.colorTools.**get_icon_color**(*filename*)
Scan an icon and return the most prevalent color.

maverig.utils.colorTools.**compare_color**(*color1*, *color2*)
Compare two colors (tuple, QColor or LabColor) with delta_e_cie_2000 human color perception approximation.

Return a float value where 2.3 corresponds to a JND (just noticeable difference).

maverig.utils.colorTools.**distinct_colors**(*base_colors*)
Return a list of colors where approximately same colors in *base_colors* are replaced by distinct colors.

maverig.utils.colorTools.**distinct_colors_from_palette**(*base_colors*, *palette_colors*)
Generate distinct colors similar to *base_colors* from *palette_colors*.

maverig.utils.colorTools.**palette_tableau10**()
Tableau-10 color set taken from http://tableaufriction.blogspot.ro/2012/11/finally-you-can-use-tableau-data-colors.html

maverig.utils.colorTools.**palette_iwanthue50**()
Generated colors with http://tools.medialab.sciences-po.fr/iwanthue/

settings: H=[0..360] C=[0.. 2] L=[0..1.15]

maverig.utils.event

class maverig.utils.event.**Event**
Bases: object

Simple Event Class - see: http://www.valuedlessons.com/2008/04/events-in-python.html usage:

handle(*handler*)

unhandle(*handler*)

demand()

fire(**args*, ***kwargs*)
fires the event with given arguments and return a list of results of each call

get_handler_count()

maverig.utils.flowlayout

class maverig.utils.flowlayout.**FlowLayout**(*parent=None*, *margin=0*, *spacing=-1*)
Bases: PySide.QtGui.QLayout

PyQt4 port of the layouts/flowlayout example from Qt v4.x

Usage Example: flowLayout = FlowLayout() flowLayout.addWidget(QtGui.QPushButton("Short")) flowLayout.addWidget(QtGui.QPushButton("Longer")) flowLayout.addWidget(QtGui.QPushButton("Different text")) flowLayout.addWidget(QtGui.QPushButton("More text")) flowLayout.addWidget(QtGui.QPushButton("Even longer button text")) self.setLayout(flowLayout)

addItem(*item*)

count()

itemAt(*index*)

takeAt(*index*)

expandingDirections()

hasHeightForWidth()

heightForWidth(*width*)

setGeometry(*rect*)

sizeHint()

minimumSize()

doLayout(*rect*, *testOnly*)

maverig.utils.forceatlas2

maverig.utils.forceatlas2.**init_edge_weights**(*g*)

maverig.utils.forceatlas2.**init_data**(*g*, *nodelist*, *dim=2*, *edge_weight=False*)
Sets the default positions and visability data edge_weight calculation should only be done if a graph already converged

maverig.utils.forceatlas2.**init_nodelist**(*g*, *nodelist=None*)

maverig.utils.forceatlas2.**length_arr**(*delta*, *minimum=0*, *squared=False*)
Return the length for each node in delta delta = [[x1,x2,x3...],[y1,y2,y3...] length_arr = [length1, length2, length3...]

delta: the numpy (dim,nnodes) array minimum: the minimum lengths to return

class maverig.utils.forceatlas2.**SpeedModel**(*mass_arr*, *ratio=0.01*, *k_s=1*, *k_s_max=10*, *max_rise=0.5*)
Bases: object

see: https://github.com/gephi/gephi/tree/master/modules/LayoutPlugin/src/main/java/org/gephi/layout/plugin/forceAtlas2/ForceAtlas.java

get_speed_arr(*force_arr*)
Return the local adaptive speed as array for each node. Call this once in each iteration after force calculation and before applying forces

class maverig.utils.forceatlas2.**ForceAtlas2**(*g*, *nodelist=None*, *attr_weight=None*, *attr_size=None*, *avoidoverlap=False*, *linlog=False*, *dissuadehubs=False*, *scale=2*, *dim=2*, *repulsion_factor=150*, *standard_size=15*)
Bases: object

Basic concept taken from: https://github.com/tpoisot/nxfa2 (Timothée Poisot). Completely revised.

A ForceAtlas2-algorithm graph layouter. Numpy optimized version with additional options, but no barnes-hut-grid approach.

do_layout()
do one layout calculation step. position data is assigned as 'pos' attribute to each node

maverig.utils.logger

class maverig.utils.logger.**StreamToLogger**(*logger*, *log_level=20*)
Bases: object

Fake file-like stream object that redirects writes to a logger instance. taken from http://www.electricmonk.nl/log/2011/08/14/redirect-stdout-and-stderr-to-a-logger-in-python

write(*buf*)

flush()

maverig.utils.logger.**activate_logger**(*log_file*)

maverig.utils.maverig_csv

class maverig.utils.maverig_csv.**CSV**
Bases: mosaik_api.Simulator

An extended copy of mosaik-csv simulator with static and dynamic data support.

Example:

House — static data — num_hh, res 3, 4 — dynamic data — Date, P # [W] 2014-10-20 00:00:00, 1080.1 2014-10-20 00:15:00, 686.06 ...

init(*sid*, *sim_start*, *datafile*)

create(*num*, *model*)

step(*time*, *inputs=None*)

get_data(*outputs*)

maverig.utils.maverig_csv.**main**()

maverig.utils.numTools

maverig.utils.numTools.**convert**(*string_value*, *to_type_str=None*)
convert string value to base type representation

maverig.utils.numTools.**get_unit_prefixed**(*value*)
return the adapted value and prefix

maverig.utils.numTools.**get_short_value_text**(*value*, *unit*)
return shortened value string with unit prefixes or exponents for small and big numbers

maverig.utils.processServer

class maverig.utils.processServer.**Call_Pack**(*func_name*, *args*, *kwargs*)
Bases: tuple

Transfer method format for calling functions via *ServerProxy*.

args
Alias for field number 1

func_name
Alias for field number 0

kwargs
Alias for field number 2

class maverig.utils.processServer.**Server**(*interval=40*)
Bases: object

A server managing incoming function calls via zero-mq socket.

register_function(*func_name*)
Register a server function that can be called via proxy.

start()
Start listening to incoming function calls and activate the socket.

run()
Repeatedly listens to incoming function calls as *Call_Pack* and trigger them.

stop()
Stop listening to incoming function calls and close socket.

class maverig.utils.processServer.**ProcessServer**(*interval=40*)
Bases: *maverig.utils.processServer.Server*

A server managing subprocesses and their incoming function calls.

register_process_factory(*process_factory*)
Register a factory method to create a subprocess.

start()
Start subprocesses and constantly listen to incoming function calls.

stop()
Stop subprocesses and stop listening to incoming functions.

class maverig.utils.processServer.**MethodProxy**(*server_proxy*, *func_name*)
Bases: object

A callable method proxy that can directly be called on the parent ServerProxy with same name and parameters like the Server function.

class maverig.utils.processServer.**ServerProxy**(*server*)
Bases: object

Subprocess (~client) side access to registered server functions. Calls can currently only be one-directional. Function return values are ommitted.

init_socket()
Activate the socket for sending method calls to *Server* from *Subprocess*. **This method must be called from Subprocess!**

close_socket()
Close the *Subprocess* socket to *Server*. **This method must be called from Subprocess!**

maverig.utils.scenarioErrors

exception maverig.utils.scenarioErrors.**ScenarioError**
Bases: Exception

title

console_text

text

info_text

elem_ids = []

exception maverig.utils.scenarioErrors.**ScenarioEmptyError**
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioOfflineElementError**(*elem_id=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioConnectionError**(*elem_id=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioBaseVoltageLevelError**(*ref_id=''*, *bus_id=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioRefBusConnectionError**(*elem_id=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioRefBusMissingError**
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioRefBusCountError**(*elem_ids=[]*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioDatafileError**(*elem_id=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioFileNotFoundError**(*elem_id=''*, *sim_name=''*, *filename=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioSimulatorError**(*elem_id=''*, *sim_name=''*, *tb=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioComponentError**(*elem_id=''*, *sim_model=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioElementError**(*elem_id=''*, *tb=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioSimulationBranchLengthError**(*elem_id=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioSimulationPowerflowError**(*elem_ids=[]*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

exception maverig.utils.scenarioErrors.**ScenarioSimulationRuntimeError**(*tb=''*)
Bases: *maverig.utils.scenarioErrors.ScenarioError*

title

console_text

text

info_text

maverig.utils.tableWidgets

class maverig.utils.tableWidgets.**CellLineEdit**(*table*, *cell_item*)
Bases: PySide.QtGui.QLineEdit

keyPressEvent(*event*)

text_changed(*text*)
add or remove rows according to last element content

class maverig.utils.tableWidgets.**AutoRowTableWidget**(**args*, ***kwargs*)
Bases: PySide.QtGui.QTableWidget

Table Widget which always leaves one empty row as last row. RowCount is automatically adapted to cell contents.

text(*row*, *col*)

row_has_text(*row*)

append_row()

maverig.utils.visSimulator

class maverig.utils.visSimulator.**VisSimulator**
Bases: mosaik_api.Simulator

The Maverig Visualization Simulator which collects simulated data from other entities and send them via *maverig.utils.processServer.ServerProxy* to maverig.models.model.SimulationServer in order to update the visualization in the main application process.

init(*sid*, *start_date*, *step_size*, *sim_proxy*, *elements*)
Initialize the simulator with the ID *sid*, the start date *start_date*, the step size *step_size*, the simulation proxy *sim_proxy* and the elements list *elements*.

create(*num*, *model*)
Create *num* instances of *model*.

step(*time*, *inputs*)
Perform the next simulation step from time *time* using input values from *inputs*, update the simulation data (time, progress, element values) via the simulation proxy and return the new simulation time (the time at which `step()` should be called again).

maverig.utils.visSimulator.**main**()

maverig.tests

- *maverig.tests.test_modePanelPresenter*
- *maverig.tests.test_consolePresenter*
- *maverig.tests.test_event*
- *maverig.tests.test_groupPresenter*
- *maverig.tests.test_menuBarPresenter*
- *maverig.tests.test_model*
- *maverig.tests.test_progressPresenter*
- *maverig.tests.test_propertyPanelPresenter*
- *maverig.tests.test_scenarioPanelPresenter*
- *maverig.tests.test_settingsPresenter*
- *maverig.tests.test_statusBarPresenter*
- *maverig.tests.test_toolbarPresenter*
- *maverig.tests.test_vPoint*

maverig.tests.test_modePanelPresenter

class maverig.tests.test_modePanelPresenter.**TestModePanelPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_remove_selected_component_and_restore_default_components()
Removes a component and restores the default components. The list of hided components in the config gets cleared.

test_hide_selected_component()
Hides a component.

test_show_invisible_components()

test_selection_mode_btn_clicked()
Switches the mode between Selection Mode and Component Mode, if the Selection Mode Button is clicked

test_hand_mode_btn_clicked()
Switches the mode between Hand Mode and Component Mode, if the Hand Mode Button is clicked

test_comp_btn_created()
Adds created button to buttons dict.

test_comp_btn_clicked()
Switches the mode between Component Mode and Selection Mode, if one Component Button is clicked

test_drag_started()
Switches the mode to component mode when a component is dragged.

test_on_change_visibility_triggered()
Triggers the visibility of the component panel

test_on_mode()
react on model mode changes and update the view buttons accordingly

test_program_mode()
react on model program mode changes

maverig.tests.test_consolePresenter

class maverig.tests.test_consolePresenter.**TestConsolePresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_on_change_visibility_triggered()
Triggers the visibility of the console panel.

test_on_console_clear_triggered()
Clear console

test_program_mode()
react on model program mode changes

test_on_output()
Appends the given output to the console output.

maverig.tests.test_event

class maverig.tests.test_event.**TestEvent**(*methodName='runTest'*)
Bases: unittest.case.TestCase

test_handle()

test_unhandle()

test_fire()

on_data_changed(*param1*, *param2*)

maverig.tests.test_groupPresenter

class maverig.tests.test_groupPresenter.**TestGroupPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_ep()
Returns the element port to a virtual point.

test_vp()
Returns the virtual point to an element port.

test_raster_snap_v_points()
Returns a list of virtual points that may snap to the raster.

test_init_scene_mapping()
Adds view items to the scene, sets related elem_id as tooltip for easier handling of errors and triggers scene mapping.

test_remove()
Removes this whole group. Unsubscribes model events.

test_snap_zone()
Returns a list of nearby virtual points (of other groups) sorted by distance (from near to far).

test_can_dock()
Returns if virtual point from_vp can dock with virtual point to_vp if docking is accepted and from_vp is not docked to another virtual point already. A virtual point (e.g. endpoint) can have one outgoing connection only but may have several ingoing connections (e.g. node).

test_to_dockables()
Returns a list of virtual points from to_vps to which the virtual point from_vp can dock-out.

test_from_dockables()
Returns a list of virtual points from from_vps to which the virtual point to_vp can dock-in.

test_connectables()
virtual points of other_vps where vp can dock-in or -out

test_new_connectables()
Returns a list of virtual points from other_vps to which the virtual point vp can dock-in or dock-out if they are not docked already.

test_non_connectable()
Returns a list of virtual points from other_vps to which the virtual point vp can't dock.

test_dock()
Docks virtual points and applies docking in model.

test_undock()
Undocks virtual points and applies undocking in model.

test_check_snap_permission()
Checks whether the view has permission to snap. Optionally validates snap restrictions.

test_on_position_changed()
Applies docking/undocking of the view within snap zone. Sets v_point position in model.

test_validation()
Updates the views validity.

test_snap_dock()
Snaps the view to the raster.

test_raster_snap()
Snaps the view to the raster if raster mode is enabled.

test_avoid_invalid_positions()
Avoids invalid positions by moving the view back to last valid positions.

test_on_mouse_released()
Applies raster snapping to the view if view is released by the mouse.

test_on_elements()
Reacts on changes on elements count and removes himself if view isn't present anymore.

test_on_positions()
Reacts on position changes.

test_on_drag()
Reacts on drag and drop.

test_on_selection()
Reacts on selection changes. Updates the views z-mode and visibility state.

test_on_dockings()
Reacts on view docking.

test_on_mode()
Reacts on mode changes. Updates the state of the view.

test_on_param()
Reacts on parameter changes.

test_change_ev_icon()
Changes the icon of an electric vehicle depending on state of charge and plugged-in state.

maverig.tests.test_menuBarPresenter

class maverig.tests.test_menuBarPresenter.**TestMenusBarPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_on_back_to_start_triggered()
Set simulated progress-visualisation to 0.

test_on_reduce_speed_triggered()
Set simulated progress-visualisation slower in speed.

test_on_run_triggered()
Runs the simulation.

test_on_stop_triggered()
Stops the simulation.

test_on_pause_triggered()
Pauses the simulation.

test_on_increase_speed_triggered()
Set simulated progress-visualisation faster in speed.

test_on_forward_to_end_triggered()
Set simulated progress-visualisation to the end of simulation.

test_on_set_time_triggered()
Sets the start time and the duration of the simulation via a dialog.

test_on_go_to_triggered()
Go to an specific simulation time position

test_on_hand_mode_triggered()
Toggles the hand mode for shifting the scenario.

test_on_selection_mode_triggered()
Toggles the hand mode for element selection.

test_on_raster_mode_triggered()
Toggles raster mode for element snapping

test_on_elements()
Reacts on changes of the elements count and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

test_on_mode()
Reacts on model changes of the current mode and toggles the checked state of the selection mode and hand mode.

test_on_drag()

test_on_vid_speed()

test_on_program_mode()
Reacts on model changes of the current program mode and adjust the ui to reflect the program mode composition, simulation or simulation paused.

test_on_selection()
Reacts on model changes of the current selection and toggles the state of the cut, copy and delete actions.

test_on_clipboard()
Reacts on model changes of the clipboard and toggles the state of the paste action.

maverig.tests.test_model

class maverig.tests.test_model.**TestModel**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_switch_modes()
Switches between the composition modes.

test_is_selectable()
Checks for an item if it is selectable or not.

test_init_history()
Init the history if an empty scene is open.

test_add_history_point()
Changes history after an action.

test_undo()
Undo the last performed action.

test_redo()
Recover the last undone action.

test_copy_to_clipboard()
Copies all elements of the elem_ids list to an internal clipboard.

test_paste_from_clipboard()
Paste all elements from internal clipboard and return the new elem_ids only dockings inside of clipboard will be maintained.

test_create_element()
Creates an element.

test_delete_element()
Deletes an element.

test_get_shared_published_params()
Return a list of composed parameter instances which are contained in each element of elem_ids, e.g.:

```
[NumResidents(), NumHouseHolds() ...]
```

- parameter.value: the parameter value of the first element of elem_ids
- parameter.shared_values: a list of values according to each element in elem_ids

test_get_param_value()
Get value of parameter with param_name in element elem_id return None if param_name does not exist.

test_get_selected()
Get value of parameter with param_name in element elem_id return None if param_name does not exist.

test_set_selected()
Sets selected-flag to marked elements.

test_docking_port()
Get the docking_port of an element.

test_get_pos()
Get the position of an element.

test_set_pos()
Set the position of an element.

test_dockings_out()
Return an array of outgoing element ports.

test_dockings_in()
Return an array of outgoing element ports.

test_can_dock()
Tests if one component may dock to another.

test_dock()
Docks branch elements to other elements, if allowed.

test_undock()
Undocks docked components.

test_update()
Fires all events with pending demands.

test_update_all()
Fires all events.

test_deselect_all_elems()
Deselect all elements.

test_copy()
Return new flat copied model including scenario, components and simulators descriptions and no connected events.

test_changes_count()
Return an int value representing the current model state changes counter.

test_get_simulator()
Return the simulator of the given element.

test_get_icon_color()
Return the color od the given icon.

test_set_param_value()
Sets a parameter of an element to an given value.

test_get_attr_values()
Return values of an attribute in a specific timestamp area.

test_get_attr_value()
Return a value of an attribute on an specific timestamp.

test_get_u_heat_value()
Return the u_heat value from an element on a specific timestamp.

test_get_i_heat_value()
Return the i_heat value from an element on a specific timestamp.

test_get_p_level()
Return the p_level from an element on a specific timestamp.

test_get_state_of_charge()
Return the state of charge from an element on a specific timestamp.

test_attr_is_multivalue()
Return True if an attribute is available in elem_ids.

test_param_is_multivalue()
Return whether parmeter values differ in elements of elem_ids.

test_docking_attrs()
Return a set of valid attribute connection tuples from one element to another.

test_handle_scenario_error()
Creates output and error events and selects elements in elem_ids for visual feedback of scenario errors.

test_validate_scenario()
Validates the scenario.

test_get_shared_published_attrs()
Return a filtered list of published attribute names which are contained in each element of elem_ids.

maverig.tests.test_progressPresenter

class maverig.tests.test_progressPresenter.**TestProgressPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_on_slider_moved()
Sets simulation data index to current slider position. Keeps slider position valid if the mosaik simulation progress isn't as far as the position. The model performs lazy updates on the UI through the refresh_timer to prevent application from speed and graph animation issues.

test_on_change_visibility_triggered()
Toggles the visibility of the progress bar. Saves the visibility state in the config.

test_on_change_dateformat()
Toggles displaying of the date.

test_on_screen_dateformat()
Reacts on changes of the date display. Displays the date as calendar date or as countdown.

test_on_progress()
Applies the current progress to the progress bar.

test_on_sim()
Reacts on simulation data index changes. Updates the slider position and the date.

test_on_program_mode()
React on model program mode changes.

test_run_slider()
Starts the progress slider.

test_stop_slider()
Stops the progress slider.

test_run_iteration()
Updates the simulation data index which is responsible for moving the progress slider.

maverig.tests.test_propertyPanelPresenter

class maverig.tests.test_propertyPanelPresenter.**TestPropertyPanelPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_on_change_visibility_triggered()
Test the hidden feature of the property panel.

test_check_spinbox()
Check that the spinbox switch to the right value.

test_value_changed()
Test if a wrong value für an element is set in the model.

test_on_selection()
Check that the right widgets are placed in the property panel when selecting an element.

test_on_param()
Check that the right widgets are placed and up to date when changing a value of an element.

test_on_program_mode()
React on model program mode changes.

maverig.tests.test_scenarioPanelPresenter

class maverig.tests.test_scenarioPanelPresenter.**TestScenarioPanelPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_adjust_scene_rect()
Fits the size of the scene to the items_bounding_rect.

test_element_at()
Returns element at mouse position.

test_group_presenters()
Returns a set of all group presenters.

test_groups()
Returns a set of all groups.

test_mouse_clicked()
Create an element on clicking left mouse button switch between selection and component mode on clicking right mouse button.

test_damped_mouse_pos()
Returns the damped mouse position if it is out of frame rect.

test_mouse_moved()
Set mouse position based on damped mouse position.

test_mouse_released()
Draw second endpoint of line or transformer.

test_zoom()
Handles zooming functionality.

test_zoom_fit()
Fits all elements into the view.

test_delete_selected_elements()
Deletes all selected elements.

test_copy_selected_elements()
Copy all selected elements.

test_cut_selected_elements()
Cut all selected elements.

test_paste_elements()
Paste and select inserted elements.

test_select_all_elements()
Selects all elements.

test_select_all_active_elements()
Selects all elements depending on the current active mode.

test_on_mode()
Reacts on model mode changes and updates the view, which component is allowed to be created and selected.

test_on_elements()
Updates the view items when number of model elements has changed.

test_run_force_layout()
Run the force atlas algorithm.

maverig.tests.test_settingsPresenter

class maverig.tests.test_settingsPresenter.**TestSettingsPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_install_language()
Sets chosen language if it is changed by user. This is handled separately to prevent handling of unnecessary events in whole application if the language hasn't been changed.

test_apply_setting()
Applies the given setting.

maverig.tests.test_statusBarPresenter

class maverig.tests.test_statusBarPresenter.**TestStatusBarPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_on_change_visibility_triggered()
Triggers the visibility of the status bar.

test_on_mode()
Reacts on mode changes. Displays the chosen mode in the status bar.

test_on_vid_speed_event()
Reacts on changes of the progress slider speed. Displays the current speed in the status bar.

test_on_program_mode()
Reacts on program mode changes. The status bar is visible in every program mode if the user didn't hide it.

test_error()
Sets the given message in the status bar.

test_info()
Sets the given message in the status bar.

test_success()
Sets the given message in the status bar.

test_reset()
Resets the state of the status bar.

maverig.tests.test_toolbarPresenter

class maverig.tests.test_toolbarPresenter.**TestToolBarPresenter**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_on_elements()
Reacts on changes of the elements count and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

test_on_selection()
Reacts on model changes of the current selection and toggles the state of the delete action.

test_on_drag()
Set the current mode

test_on_sim()
Reacts on changes of the simulation time and speed parameters and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

test_on_vid_speed()
Reacts on changes of the progress slider speed and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

test_on_program_mode()
Reacts on program mode changes and toggles the state (checked/unchecked/enabled/disabled) of the depending actions.

maverig.tests.test_vPoint

class maverig.tests.test_vPoint.**TestVPoint**(*methodName='runTest'*)
Bases: unittest.case.TestCase

setUp()

test_set_pos()

test_move_pos()

test_follow_follows()
self keeps hold of v_point. adjust self when v_point position change applies to trigger. self will change it's position with given reason

test_unfollow()

test_fix()
self keeps hold of v_point and vice versa

test_unfix()

maverig.EntryPoint

class maverig.EntryPoint.**EntryPoint**
Bases: object

The maverig application class which creates and connects model, views and presenters.

maverig.EntryPoint.**main**()
Starts the maverig application with an empty scenario.

maverig.demo

maverig.demo.**main**()
Starts the maverig application with a demo scenario.

CHAPTER 6

About Maverig

Maverig is a Graphical User Interface for creation and visualization of Smart-Grid simulations using the mosaik framework. Maverig has been developed by a project group of students from the Carl von Ossietzky University Oldenburg, Germany in cooperation with OFFIS.

Note Maverig is currently in a prototypical state. It may be used for demonstration purposes, but its maintenance and further development is not our top priority right now. Please understand, if the installation guide is not up-to-date with the newest package versions.

Contributors

Erika Root, Gerrit Klasen, Hanno Günther, Jerome Tammen, Marina Sartison, Marius Brinkmann, Michael Falk, Rafael Burschik, Rouven Pajewski Sascha Spengler, Andrianarisoa A. Johary Ny Aina and Tobias Schwerdtfeger

CHAPTER 7

Indices and tables

- genindex
- modindex
- search

m

maverig.data.config, 16
maverig.data.dataHandler, 18
maverig.data.settings.abstractSettings, 15
maverig.data.settings.defaultSettings, 15
maverig.data.settings.heatValueEffect, 16
maverig.data.settings.settings, 16
maverig.demo, 73
maverig.EntryPoint, 73
maverig.models.model, 19
maverig.models.modelGraph, 26
maverig.models.modelSimulation, 26
maverig.presenter.abstractPresenter, 31
maverig.presenter.attributePanelPresenter, 31
maverig.presenter.componentWizardPresenter, 33
maverig.presenter.consolePanelPresenter, 33
maverig.presenter.group_presenter.abstractGroupPresenter, 28
maverig.presenter.group_presenter.iconGroupPresenter, 30
maverig.presenter.group_presenter.lineGroupPresenter, 30
maverig.presenter.group_presenter.lineIconGroupPresenter, 30
maverig.presenter.group_presenter.nodeGroupPresenter, 30
maverig.presenter.menuBarPresenter, 34
maverig.presenter.modePanelPresenter, 32
maverig.presenter.presenterManager, 36
maverig.presenter.progressPresenter, 37
maverig.presenter.propertyPanelPresenter, 37
maverig.presenter.scenarioPanelPresenter, 38
maverig.presenter.settingsPresenter, 39
maverig.presenter.statusBarPresenter, 40
maverig.presenter.toolbarPresenter, 40
maverig.presenter.utils.forceEngine, 31
maverig.tests.test_consolePresenter, 64
maverig.tests.test_event, 64
maverig.tests.test_groupPresenter, 64
maverig.tests.test_menuBarPresenter, 66
maverig.tests.test_model, 67
maverig.tests.test_modePanelPresenter, 63
maverig.tests.test_progressPresenter, 69
maverig.tests.test_propertyPanelPresenter, 70
maverig.tests.test_scenarioPanelPresenter, 70
maverig.tests.test_settingsPresenter, 71
maverig.tests.test_statusBarPresenter, 72
maverig.tests.test_toolbarPresenter, 72
maverig.tests.test_vPoint, 73
maverig.utils.colorTools, 55
maverig.utils.event, 56
maverig.utils.flowlayout, 56
maverig.utils.forceatlas2, 57
maverig.utils.logger, 57
maverig.utils.maverig_csv, 58
maverig.utils.numTools, 58
maverig.utils.processServer, 58
maverig.utils.scenarioErrors, 59
maverig.utils.tableWidgets, 62
maverig.utils.visSimulator, 62
maverig.views.abstractView, 49
maverig.views.attributePanelView, 49
maverig.views.componentWizardView, 51
maverig.views.consolePanelView, 52
maverig.views.dialogs, 52

maverig.views.groups.abstractGroup, 42
maverig.views.groups.iconGroup, 43
maverig.views.groups.lineGroup, 43
maverig.views.groups.lineIconGroup, 43
maverig.views.groups.nodeGroup, 43
maverig.views.items.abstractItem, 43
maverig.views.items.circle, 44
maverig.views.items.icon, 44
maverig.views.items.line, 45
maverig.views.mainWindow, 52
maverig.views.menuBarView, 53
maverig.views.modePanelView, 50
maverig.views.positioning.section, 45
maverig.views.positioning.vPoint, 47
maverig.views.progressView, 53
maverig.views.propertyPanelView, 53
maverig.views.scenarioPanelView, 54
maverig.views.settingsView, 54
maverig.views.statusBarView, 55
maverig.views.toolbarView, 55

A

about_dialog() (in module maverig.views.dialogs), 52
AbstractGroup (class in maverig.views.groups.abstractGroup), 42
AbstractGroupPresenter (class in maverig.presenter.group_presenter.abstractGroupPresenter), 28
AbstractItem (class in maverig.views.items.abstractItem), 43
AbstractPresenter (class in maverig.presenter.abstractPresenter), 31
AbstractView (class in maverig.views.abstractView), 49
activate_logger() (in module maverig.utils.logger), 58
ACTIVATED_AUTO_LAYOUT_MODE() (in module maverig.data.config), 18
ACTIVATED_COMPONENT_MODE() (in module maverig.data.config), 18
ACTIVATED_HAND_MODE() (in module maverig.data.config), 18
ACTIVATED_SELECTION_MODE() (in module maverig.data.config), 18
ACTIVATED_SIMULATION_MODE() (in module maverig.data.config), 18
add_component_btn_clicked() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
add_endpoint() (maverig.views.groups.iconGroup.IconGroup method), 43
add_history_point() (maverig.models.model.Model method), 23
add_item() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
add_new_attribute_tab() (maverig.views.componentWizardView.AttributeParameterPage method), 51
add_new_icon() (maverig.views.componentWizardView.IntroPage method), 51
add_new_parameter_tab() (maverig.views.componentWizardView.AttributeParameterPage method), 51
add_property_value_object() (maverig.views.propertyPanelView.PropertyPanelView method), 53
add_to_scene() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
add_to_scene() (maverig.views.items.abstractItem.AbstractItem method), 43
add_v_point() (maverig.views.items.abstractItem.AbstractItem method), 43
addItem() (maverig.utils.flowlayout.FlowLayout method), 56
adjust_line() (maverig.views.items.line.Line method), 45
adjust_scene_rect() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
all (maverig.views.positioning.vPoint.Changes attribute), 47
alter_boolean_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
alter_file_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
alter_integer_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
append_row() (maverig.utils.tableWidgets.AutoRowTableWidget method), 62
applied (maverig.views.positioning.vPoint.Change attribute), 47
apply_positions() (maverig.presenter.utils.forceEngine.ForceEngine method), 31
apply_setting() (maverig.presenter.settingsPresenter.SettingsPresenter method), 40
apply_settings() (maverig.presenter.settingsPresenter.SettingsPresenter method), 40
args (maverig.utils.processServer.Call_Pack attribute), 58
attr_is_multivalue() (maverig.models.model.Model method), 24
ATTRIBUTE_GRAPHS_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
AttributeCell (class in

maverig.views.attributePanelView), 49
AttributePanelPresenter (class in maverig.presenter.attributePanelPresenter), 31
AttributePanelView (class in maverig.views.attributePanelView), 49
AttributeParameterPage (class in maverig.views.componentWizardView), 51
AttributeTab (class in maverig.views.componentWizardView), 51
auto_update_components_languages (maverig.models.model.Model attribute), 20
AutoRowTableWidget (class in maverig.utils.tableWidgets), 62
avoid_invalid (maverig.views.positioning.vPoint.Change attribute), 47
avoid_invalid_positions() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29

B

button_mouse_move() (maverig.views.modePanelView.ModePanelView method), 50
button_mouse_pressed() (maverig.views.modePanelView.ModePanelView method), 50

C

calculated (maverig.views.positioning.vPoint.Change attribute), 47
Call_Pack (class in maverig.utils.processServer), 58
can_dock() (maverig.models.model.Model method), 25
can_dock() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
category_update() (maverig.views.componentWizardView.IntroPage method), 51
CellLineEdit (class in maverig.utils.tableWidgets), 62
Change (class in maverig.views.positioning.vPoint), 47
change (maverig.views.positioning.vPoint.VPoint attribute), 48
change_color() (maverig.views.propertyPanelView.PropertyPanelView method), 53
change_content() (maverig.views.attributePanelView.AttributeCell method), 50
change_ev_icon() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
change_household_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
Changes (class in maverig.views.positioning.vPoint), 47
changes_count() (maverig.models.model.Model method), 23
CHECK_BOX (maverig.data.settings.settings.SettingTypes attribute), 16
check_snap_permission() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
check_spinbox() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 38
check_state() (maverig.views.componentWizardView.AttributeTab method), 51
check_state() (maverig.views.componentWizardView.IntroPage method), 51
check_state() (maverig.views.componentWizardView.ParameterTab method), 51
CheckBoxSetting (class in maverig.data.settings.abstractSettings), 15
Circle (class in maverig.views.items.circle), 44
circle_style (maverig.views.items.circle.Circle attribute), 44
clear_container() (maverig.views.attributePanelView.AttributePanelView method), 49
clear_effects() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
clear_effects() (maverig.views.items.abstractItem.AbstractItem method), 44
clear_prop_grid() (maverig.views.propertyPanelView.PropertyPanelView method), 53
clear_state_of_charge_effect() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
clear_state_of_charge_effect() (maverig.views.items.abstractItem.AbstractItem method), 44
clear_state_of_charge_tip() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
clear_state_of_charge_tip() (maverig.views.items.abstractItem.AbstractItem method), 44
clear_state_of_charge_tip_bg() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
clear_state_of_charge_tip_bg() (maverig.views.items.abstractItem.AbstractItem method), 44
clipboard_elements (maverig.models.model.Model attribute), 21
close_socket() (maverig.utils.processServer.ServerProxy method), 59
closeEvent() (maverig.views.mainWindow.MainWindow method), 52
color_interp() (in module maverig.utils.colorTools), 55
COMBO_BOX (maverig.data.settings.settings.SettingTypes attribute), 16
ComboBoxSetting (class in maverig.data.settings.abstractSettings), 15
ComboUpdatingBox (class in

maverig.views.componentWizardView), 52
comp (maverig.models.model.Mode attribute), 20
comp (maverig.models.model.Model attribute), 23
comp_btn_clicked() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
comp_btn_created() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
comp_raster (maverig.models.model.Model attribute), 23
compare_color() (in module maverig.utils.colorTools), 56
components (maverig.models.model.Model attribute), 22
ComponentWizardPresenter (class in maverig.presenter.componentWizardPresenter), 33
composition (maverig.models.model.ProgramMode attribute), 20
ConclusionPage (class in maverig.views.componentWizardView), 51
ConfigKeys (class in maverig.data.config), 17
connectables() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
console_text (maverig.utils.scenarioErrors.ScenarioBaseVoltageLevelError attribute), 60
console_text (maverig.utils.scenarioErrors.ScenarioComponentError attribute), 61
console_text (maverig.utils.scenarioErrors.ScenarioConnectionError attribute), 60
console_text (maverig.utils.scenarioErrors.ScenarioDatafileError attribute), 61
console_text (maverig.utils.scenarioErrors.ScenarioElementError attribute), 61
console_text (maverig.utils.scenarioErrors.ScenarioEmptyError attribute), 60
console_text (maverig.utils.scenarioErrors.ScenarioError attribute), 59
console_text (maverig.utils.scenarioErrors.ScenarioFileNotFoundError attribute), 61
console_text (maverig.utils.scenarioErrors.ScenarioOfflineElementError attribute), 60
console_text (maverig.utils.scenarioErrors.ScenarioRefBusConnectionError attribute), 60
console_text (maverig.utils.scenarioErrors.ScenarioRefBusCountError attribute), 61
console_text (maverig.utils.scenarioErrors.ScenarioRefBusMissingError attribute), 60
console_text (maverig.utils.scenarioErrors.ScenarioSimulationBranchLengthError attribute), 62
console_text (maverig.utils.scenarioErrors.ScenarioSimulationPowerflowError attribute), 62
console_text (maverig.utils.scenarioErrors.ScenarioSimulationRuntimeError attribute), 62
console_text (maverig.utils.scenarioErrors.ScenarioSimulatorError attribute), 61
ConsolePanelPresenter (class in maverig.presenter.consolePanelPresenter), 33
ConsolePanelView (class in maverig.views.consolePanelView), 52
contextMenuPresenter() (maverig.views.modePanelView.ModePanelView method), 50
contextMenuPresenter() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
convert() (in module maverig.utils.numTools), 58
convert_slider_value() (maverig.views.dialogs.SimulationTimeDialog method), 52
copy() (maverig.models.model.Model method), 24
copy_selected_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresente method), 39
copy_to_clipboard() (maverig.models.model.Model method), 24
count() (maverig.utils.flowlayout.FlowLayout method), 56
create() (maverig.utils.visSimulator.VisSimulator method), 63
create_attribute_cell() (maverig.views.attributePanelView.AttributePanelVie method), 49
create_attribute_panel() (maverig.views.attributePanelView.AttributePanelV method), 49
create_boolean_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_btn_context_menu() (maverig.views.modePanelView.ModePanelView method), 50
create_button() (maverig.views.modePanelView.ModePanelView method), 50
create_category_layout() (maverig.views.modePanelView.ModePanelView method), 50
create_components_grid() (maverig.views.modePanelView.ModePanelView method), 50
create_components_language_po_entries() (in module maverig.data.config), 18
create_context_menu() (maverig.views.modePanelView.ModePanelView method), 51
create_context_menu() (maverig.views.scenarioPanelView.ScenarioPanelVi method), 54
create_element() (maverig.models.model.Model method), 24
create_file_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_float_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_graph() (maverig.views.attributePanelView.AttributeCell

method), 49
create_household_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_integer_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_new_element() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
create_property_icon() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_property_label() (maverig.views.propertyPanelView.PropertyPanelView method), 53
create_str_property_cell() (maverig.views.propertyPanelView.PropertyPanelView method), 53
CREATION_INVALID() (in module maverig.data.config), 18
CSV (class in maverig.utils.maverig_csv), 58
cut_selected_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39

D

damped_mouse_pos() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
datetime_to_qdatetime() (maverig.presenter.menuBarPresenter.MenuBarPresenter static method), 36
day_night_vis_setting (maverig.data.settings.settings.Settings attribute), 16
delete_element() (maverig.models.model.Model method), 24
delete_grid_widget() (maverig.views.propertyPanelView.PropertyPanelView method), 53
delete_selected_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
delta (maverig.views.positioning.vPoint.VPoint attribute), 47
demand() (maverig.utils.event.Event method), 56
deselect_all_elems() (maverig.models.model.Model method), 26
distinct_colors() (in module maverig.utils.colorTools), 56
distinct_colors_from_palette() (in module maverig.utils.colorTools), 56
do_layout() (maverig.utils.forceatlas2.ForceAtlas2 method), 57
dock() (maverig.models.model.Model method), 25
dock() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
docking_attrs() (maverig.models.model.Model method), 25
DOCKING_INVALID() (in module maverig.data.config), 18
DOCKING_NO_ITEMS() (in module maverig.data.config), 18
docking_port() (maverig.models.model.Model method), 25
DOCKING_VALID() (in module maverig.data.config), 18
dockings_in() (maverig.models.model.Model method), 25
dockings_out() (maverig.models.model.Model method), 25
dropEvent() (maverig.utils.flowlayout.FlowLayout method), 57
dropEvent() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
dragEnterEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
dragMoveEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
draw_canvas() (maverig.views.attributePanelView.AttributeCell method), 50
drawForeground() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
drawBackground() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
dropEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
duration (maverig.models.model.Model attribute), 22

E

EFFECT_BAR (maverig.data.settings.heatValueEffect.HeatValueEffectKey attribute), 16
EFFECT_BLUR (maverig.data.settings.heatValueEffect.HeatValueEffectKey attribute), 16
EFFECT_COLOR (maverig.data.settings.heatValueEffect.HeatValueEffectKey attribute), 16
EFFECT_SHADOW (maverig.data.settings.heatValueEffect.HeatValueEffectKey attribute), 16
EFFECT_TRANSPARENCY (maverig.data.settings.heatValueEffect.HeatValueEffectKeys attribute), 16
elem_id (maverig.models.model.ElemPort attribute), 19
elem_ids (maverig.utils.scenarioErrors.ScenarioError attribute), 59
elem_ports() (maverig.models.model.Model method), 25
element_already_exist() (in module maverig.views.dialogs), 52
element_at() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
elements (maverig.models.model.Model attribute), 20
ElemPort (class in maverig.models.model), 19
enabled (maverig.views.groups.abstractGroup.AbstractGroup attribute), 42
enabled (maverig.views.items.abstractItem.AbstractItem attribute), 44
ensure_dir() (in module maverig.data.dataHandler), 18

enter() (maverig.views.positioning.section.Section method), 45
EntryPoint (class in maverig.EntryPoint), 73
ep() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
error() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
error_dialog() (in module maverig.views.dialogs), 52
Event (class in maverig.utils.event), 56
exit() (maverig.views.positioning.section.Section method), 45
expandingDirections() (maverig.utils.flowlayout.FlowLayout method), 56
extern_followers (maverig.views.positioning.vPoint.VPoint attribute), 48

F

fast_deepcopy() (in module maverig.models.model), 19
FILE_OPENED() (in module maverig.data.config), 18
FILE_SAVED() (in module maverig.data.config), 18
fire() (maverig.utils.event.Event method), 56
fix() (maverig.views.positioning.vPoint.VPoint method), 48
FlowLayout (class in maverig.utils.flowlayout), 56
flush() (maverig.utils.logger.StreamToLogger method), 58
follow() (maverig.views.positioning.vPoint.VPoint method), 48
followed (maverig.views.positioning.vPoint.Change attribute), 47
followers (maverig.views.positioning.vPoint.VPoint attribute), 48
follows() (maverig.views.positioning.vPoint.VPoint method), 48
force_dragging (maverig.models.model.Model attribute), 23
ForceAtlas2 (class in maverig.utils.forceatlas2), 57
ForceEngine (class in maverig.presenter.utils.forceEngine), 31
from_dockables() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
func_name (maverig.utils.processServer.Call_Pack attribute), 58

G

GENERAL_SETTINGS (maverig.data.config.ConfigKeys attribute), 17
general_settings_tab (maverig.data.settings.settings.Settings attribute), 16
get_attr_value() (maverig.models.model.Model method), 24
get_attr_values() (maverig.models.model.Model method), 24
get_category_names() (maverig.presenter.componentWizardPresenter.ComponentWizardPresenter method), 33
get_component() (maverig.models.model.Model method), 24
get_component_icon() (in module maverig.data.dataHandler), 19
get_config_file() (in module maverig.data.dataHandler), 19
get_data() (maverig.utils.maverig_csv.CSV method), 58
get_handler_count() (maverig.utils.event.Event method), 56
get_i_heat_value() (maverig.models.model.Model method), 25
get_icon() (in module maverig.data.dataHandler), 19
get_icon_color() (in module maverig.utils.colorTools), 55
get_icon_color() (maverig.models.model.Model method), 24
get_lang_path() (in module maverig.data.dataHandler), 19
get_maverig_dir() (in module maverig.data.dataHandler), 18
get_normpath() (in module maverig.data.dataHandler), 18
get_p_level() (maverig.models.model.Model method), 25
get_param_value() (maverig.models.model.Model method), 24
get_pos() (maverig.models.model.Model method), 25
get_published_components() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 33
get_relpath() (in module maverig.data.dataHandler), 18
get_selected() (maverig.models.model.Model method), 25
get_shared_published_attrs() (maverig.models.model.Model method), 24
get_shared_published_params() (maverig.models.model.Model method), 24
get_short_value_text() (in module maverig.utils.numTools), 58
get_simulator() (maverig.models.model.Model method), 24
get_simulator_names() (maverig.presenter.componentWizardPresenter.ComponentWizardPresenter method), 33
get_speed_arr() (maverig.utils.forceatlas2.SpeedModel method), 57
get_state_of_charge() (maverig.models.model.Model method), 25
get_temp_file() (in module maverig.data.dataHandler), 19
get_u_heat_value() (maverig.models.model.Model method), 25
get_unit_prefixed() (in module maverig.utils.numTools), 58
go_to_time_dialog() (in module maverig.views.dialogs),

52
graph (maverig.models.model.Model attribute), 21
group_presenters (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter attribute), 39
groups (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter attribute), 39

H

hand (maverig.models.model.Mode attribute), 20
hand_mode_btn_clicked() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
handle() (maverig.utils.event.Event method), 56
handle_process_scenario_error() (maverig.models.modelSimulation.SimulationServer method), 27
handle_scenario_error() (maverig.models.model.Model method), 26
hasHeightForWidth() (maverig.utils.flowlayout.FlowLayout method), 57
HEAT_VALUE_EFFECT_CPP (maverig.data.config.ConfigKeys attribute), 17
HEAT_VALUE_EFFECT_GRIDS (maverig.data.config.ConfigKeys attribute), 17
heat_value_effect_setting1 (maverig.data.settings.settings.Settings attribute), 16
heat_value_effect_setting2 (maverig.data.settings.settings.Settings attribute), 16
HeatValueEffectKeys (class in maverig.data.settings.heatValueEffect), 16
heightForWidth() (maverig.utils.flowlayout.FlowLayout method), 57
hide_selected_component() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
history_redo (maverig.models.model.Model attribute), 21
history_undo (maverig.models.model.Model attribute), 21
hover_component_button() (maverig.views.modePanelView.ModePanelView method), 50

I

Icon (class in maverig.views.items.icon), 44
icon_path (maverig.views.items.icon.Icon attribute), 44
icon_update (maverig.views.componentWizardView.IntroPage attribute), 51
IconGroup (class in maverig.views.groups.iconGroup), 43
IconGroupPresenter (class in maverig.presenter.group_presenter.iconGroupPresenter), 39
import_method() (in module maverig.data.config), 16
info (maverig.views.positioning.vPoint.Changes attribute), 47
info() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
info_text (maverig.utils.scenarioErrors.ScenarioBaseVoltageLevelError attribute), 60
info_text (maverig.utils.scenarioErrors.ScenarioComponentError attribute), 61
info_text (maverig.utils.scenarioErrors.ScenarioConnectionError attribute), 60
info_text (maverig.utils.scenarioErrors.ScenarioDatafileError attribute), 61
info_text (maverig.utils.scenarioErrors.ScenarioElementError attribute), 61
info_text (maverig.utils.scenarioErrors.ScenarioEmptyError attribute), 60
info_text (maverig.utils.scenarioErrors.ScenarioError attribute), 59
info_text (maverig.utils.scenarioErrors.ScenarioFileNotFoundError attribute), 61
info_text (maverig.utils.scenarioErrors.ScenarioOfflineElementError attribute), 60
info_text (maverig.utils.scenarioErrors.ScenarioRefBusConnectionError attribute), 60
info_text (maverig.utils.scenarioErrors.ScenarioRefBusCountError attribute), 61
info_text (maverig.utils.scenarioErrors.ScenarioRefBusMissingError attribute), 60
info_text (maverig.utils.scenarioErrors.ScenarioSimulationBranchLengthEr attribute), 62
info_text (maverig.utils.scenarioErrors.ScenarioSimulationPowerflowError attribute), 62
info_text (maverig.utils.scenarioErrors.ScenarioSimulationRuntimeError attribute), 62
info_text (maverig.utils.scenarioErrors.ScenarioSimulatorError attribute), 61
inform_dialog() (in module maverig.views.dialogs), 52
init() (maverig.utils.maverig_csv.CSV method), 58
init() (maverig.utils.visSimulator.VisSimulator method), 62
init_data() (in module maverig.utils.forceatlas2), 57
init_edge_weights() (in module maverig.utils.forceatlas2), 57
init_graphics_item() (maverig.views.items.abstractItem.AbstractItem method), 43
init_graphics_item() (maverig.views.items.circle.Circle method), 44
init_graphics_item() (maverig.views.items.icon.Icon method), 44
init_graphics_item() (maverig.views.items.line.Line

method), 45
init_history() (maverig.models.model.Model method), 23
init_nodelist() (in module maverig.utils.forceatlas2), 57
init_scene_mapping() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
init_selection_counter() (maverig.views.propertyPanelView.PropertyPanelView method), 53
init_socket() (maverig.utils.processServer.ServerProxy method), 59
init_ui() (maverig.views.attributePanelView.AttributePanelView method), 49
init_ui() (maverig.views.consolePanelView.ConsolePanelView method), 52
init_ui() (maverig.views.menuBarView.MenuBarView method), 53
init_ui() (maverig.views.modePanelView.ModePanelView method), 50
init_ui() (maverig.views.progressView.ProgressView method), 53
init_ui() (maverig.views.propertyPanelView.PropertyPanelView method), 53
init_ui() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
init_ui() (maverig.views.statusBarView.StatusBarView method), 55
init_ui() (maverig.views.toolbarView.ToolbarView method), 55
init_view() (maverig.presenter.componentWizardPresenter.ComponentWizardPresenter method), 33
init_view() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
initializePage() (maverig.views.componentWizardView.ConclusionPage method), 51
INPUT (maverig.data.settings.settings.SettingTypes attribute), 16
InputSetting (class in maverig.data.settings.abstractSettings), 15
install_language() (maverig.presenter.settingsPresenter.SettingsPresenter method), 40
IntroPage (class in maverig.views.componentWizardView), 51
INVISIBLE_COMPONENTS (maverig.data.config.ConfigKeys attribute), 18
is_acceptable() (in module maverig.views.componentWizardView), 51
IS_ATTRIBUTE_PANEL_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
IS_COMPONENT_PANEL_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
IS_CONSOLE_PANEL_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
IS_DAY_NIGHT_VIS_ENABLED (maverig.data.config.ConfigKeys attribute), 17
IS_HEAT_VALUE_EFFECT_FOR_CPP_ENABLED (maverig.data.config.ConfigKeys attribute), 17
IS_HEAT_VALUE_EFFECT_FOR_GRIDS_ENABLED (maverig.data.config.ConfigKeys attribute), 17
is_ok() (maverig.views.componentWizardView.AttributeTab method), 52
is_ok() (maverig.views.componentWizardView.ParameterTab method), 51
IS_PROGRESS_BAR_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
IS_PROPERTY_PANEL_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
is_running() (maverig.views.positioning.section.Section method), 45
is_selectable() (maverig.models.model.Model method), 23
IS_STATUS_BAR_VISIBLE (maverig.data.config.ConfigKeys attribute), 17
is_under_mouse (maverig.views.groups.abstractGroup.AbstractGroup attribute), 43
is_under_mouse (maverig.views.items.abstractItem.AbstractItem attribute), 44
isComplete() (maverig.views.componentWizardView.AttributeParameterPage method), 51
isComplete() (maverig.views.componentWizardView.IntroPage method), 51
itemAt() (maverig.utils.flowlayout.FlowLayout method), 56

K

keyPressEvent() (maverig.utils.tableWidgets.CellLineEdit method), 62
kwargs (maverig.utils.processServer.Call_Pack attribute), 58

L

lab_color() (in module maverig.utils.colorTools), 55
LANGUAGE (maverig.data.config.ConfigKeys attribute), 17
language (maverig.models.model.Model attribute), 22
languages (maverig.data.settings.settings.Settings attribute), 16
last_positions (maverig.views.positioning.vPoint.VPoint attribute), 47
length_arr() (in module maverig.utils.forceatlas2), 57
Line (class in maverig.views.items.line), 45
line_style (maverig.views.items.line.Line attribute), 45

LINE_TOO_SHORT() (in module maverig.data.config), 18
LineGroup (class in maverig.views.groups.lineGroup), 43
LineGroupPresenter (class in maverig.presenter.group_presenter.lineGroupPresenter), 30
LineIconGroup (class in maverig.views.groups.lineIconGroup), 43

M

main() (in module maverig.demo), 73
main() (in module maverig.EntryPoint), 73
main() (in module maverig.utils.maverig_csv), 58
main() (in module maverig.utils.visSimulator), 63
MAIN_WINDOW_GEOMETRY (maverig.data.config.ConfigKeys attribute), 17
MAIN_WINDOW_STATE (maverig.data.config.ConfigKeys attribute), 17
MainWindow (class in maverig.views.mainWindow), 52
map_elem_to_mosaik() (maverig.models.modelSimulation.SimulationServer method), 27
mappings_port_vp (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter attribute), 28
mappings_port_vp (maverig.presenter.group_presenter.iconGroupPresenter.IconGroupPresenter attribute), 30
mappings_port_vp (maverig.presenter.group_presenter.lineGroupPresenter.LineGroupPresenter attribute), 30
mappings_port_vp (maverig.presenter.group_presenter.nodeGroupPresenter.NodeGroupPresenter attribute), 30
maverig.data.config (module), 16
maverig.data.dataHandler (module), 18
maverig.data.settings.abstractSettings (module), 15
maverig.data.settings.defaultSettings (module), 15
maverig.data.settings.heatValueEffect (module), 16
maverig.data.settings.settings (module), 16
maverig.demo (module), 73
maverig.EntryPoint (module), 73
maverig.models.model (module), 19
maverig.models.modelGraph (module), 26
maverig.models.modelSimulation (module), 26
maverig.presenter.abstractPresenter (module), 31
maverig.presenter.attributePanelPresenter (module), 31
maverig.presenter.componentWizardPresenter (module), 33
maverig.presenter.consolePanelPresenter (module), 33
maverig.presenter.group_presenter.abstractGroupPresenter (module), 28
maverig.presenter.group_presenter.iconGroupPresenter (module), 30
maverig.presenter.group_presenter.lineGroupPresenter (module), 30
maverig.presenter.group_presenter.lineIconGroupPresenter (module), 30
maverig.presenter.group_presenter.nodeGroupPresenter (module), 30
maverig.presenter.menuBarPresenter (module), 34
maverig.presenter.modePanelPresenter (module), 32
maverig.presenter.presenterManager (module), 36
maverig.presenter.progressPresenter (module), 37
maverig.presenter.propertyPanelPresenter (module), 37
maverig.presenter.scenarioPanelPresenter (module), 38
maverig.presenter.settingsPresenter (module), 39
maverig.presenter.statusBarPresenter (module), 40
maverig.presenter.toolbarPresenter (module), 40
maverig.presenter.utils.forceEngine (module), 31
maverig.tests.test_consolePresenter (module), 64
maverig.tests.test_event (module), 64
maverig.tests.test_groupPresenter (module), 64
maverig.tests.test_menuBarPresenter (module), 66
maverig.tests.test_model (module), 67
maverig.tests.test_modePanelPresenter (module), 63
maverig.tests.test_progressPresenter (module), 69
maverig.tests.test_propertyPanelPresenter (module), 70
maverig.tests.test_scenarioPanelPresenter (module), 70
maverig.tests.test_settingsPresenter (module), 71
maverig.tests.test_statusBarPresenter (module), 72
maverig.tests.test_toolbarPresenter (module), 72
maverig.tests.test_vPoint (module), 73
maverig.utils.colorTools (module), 55
maverig.utils.event (module), 56
maverig.utils.flowlayout (module), 56
maverig.utils.forceatlas2 (module), 57
maverig.utils.logger (module), 57
maverig.utils.maverig_csv (module), 58
maverig.utils.numTools (module), 58
maverig.utils.processServer (module), 58
maverig.utils.scenarioErrors (module), 59
maverig.utils.tableWidgets (module), 62
maverig.utils.visSimulator (module), 62
maverig.views.abstractView (module), 49
maverig.views.attributePanelView (module), 49
maverig.views.componentWizardView (module), 51
maverig.views.consolePanelView (module), 52
maverig.views.dialogs (module), 52
maverig.views.groups.abstractGroup (module), 42
maverig.views.groups.iconGroup (module), 43
maverig.views.groups.lineGroup (module), 43
maverig.views.groups.lineIconGroup (module), 43
maverig.views.groups.nodeGroup (module), 43
maverig.views.items.abstractItem (module), 43
maverig.views.items.circle (module), 44
maverig.views.items.icon (module), 44
maverig.views.items.line (module), 45
maverig.views.mainWindow (module), 52
maverig.views.menuBarView (module), 53

maverig.views.modePanelView (module), 50
maverig.views.positioning.section (module), 45
maverig.views.positioning.vPoint (module), 47
maverig.views.progressView (module), 53
maverig.views.propertyPanelView (module), 53
maverig.views.scenarioPanelView (module), 54
maverig.views.settingsView (module), 54
maverig.views.statusBarView (module), 55
maverig.views.toolbarView (module), 55
MenuBarPresenter (class in maverig.presenter.menuBarPresenter), 34
MenuBarView (class in maverig.views.menuBarView), 53
MethodProxy (class in maverig.utils.processServer), 59
minimumSize() (maverig.utils.flowlayout.FlowLayout method), 57
Mode (class in maverig.models.model), 20
mode (maverig.models.model.Model attribute), 23
MODE_PANEL_SETTINGS (maverig.data.config.ConfigKeys attribute), 17
Model (class in maverig.models.model), 20
ModelGraph (class in maverig.models.modelGraph), 26
ModePanelPresenter (class in maverig.presenter.modePanelPresenter), 32
ModePanelView (class in maverig.views.modePanelView), 50
mouse_clicked() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
mouse_moved() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
mouse_released() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
mouseDoubleClickEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
mouseMoveEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
mousePressEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
mouseReleaseEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
move_pos() (maverig.views.items.abstractItem.AbstractItem method), 44
move_pos() (maverig.views.items.circle.Circle method), 44
move_pos() (maverig.views.items.icon.Icon method), 44
move_pos() (maverig.views.items.line.Line method), 45
move_pos() (maverig.views.positioning.vPoint.VPoint method), 48
moved (maverig.views.positioning.vPoint.Change attribute), 47
MULTI_SELECT_DIFFERENT_VALUES() (in module maverig.data.config), 18

N

new_category_dialog() (maverig.views.componentWizardView.IntroPage method), 51
new_connectables() (maverig.presenter.group_presenter.abstractGroupPrese method), 28
new_file() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
new_pos (maverig.views.positioning.vPoint.VPoint attribute), 47
NodeGroup (class in maverig.views.groups.nodeGroup), 43
NodeGroupPresenter (class in maverig.presenter.group_presenter.nodeGroupPresenter), 30
non_connectable() (maverig.presenter.group_presenter.abstractGroupPresen method), 29
none (maverig.views.positioning.vPoint.Changes attribute), 47
np_color() (in module maverig.utils.colorTools), 55

O

on_about_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresen static method), 36
on_add_component() (maverig.presenter.componentWizardPresenter.Compo method), 33
on_add_simulator_triggered() (maverig.presenter.componentWizardPresenter.ComponentWizard method), 33
on_attrs() (maverig.presenter.attributePanelPresenter.AttributePanelPresente method), 32
on_attrs() (maverig.presenter.group_presenter.abstractGroupPresenter.Abstr method), 29
on_attrs() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresente method), 39
on_auto_layout_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_auto_layout_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_back_to_start_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_back_to_start_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_btn_context_menu() (maverig.presenter.modePanelPresenter.ModePanel method), 32
on_change_dateformat() (maverig.presenter.progressPresenter.ProgressPres method), 37
on_change_graph_visibility_triggered() (maverig.presenter.attributePanelPresenter.AttributePanelPresente

method), 31
on_change_visibility_triggered() (maverig.presenter.attributePanelPresenter.AttributePanelPresenter method), 31
on_change_visibility_triggered() (maverig.presenter.consolePanelPresenter.ConsolePanelPresenter method), 33
on_change_visibility_triggered() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
on_change_visibility_triggered() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_change_visibility_triggered() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 37
on_change_visibility_triggered() (maverig.presenter.settingsPresenter.SettingsPresenter method), 39
on_change_visibility_triggered() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
on_clipboard() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_close_wizard() (maverig.presenter.componentWizardPresenter.ComponentWizardPresenter method), 33
on_components() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 33
on_console_clear_triggered() (maverig.presenter.consolePanelPresenter.ConsolePanelPresenter method), 33
on_context_menu() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
on_context_menu() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
on_copy_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_cut_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_data_changed() (maverig.tests.test_event.TestEvent method), 64
on_delete_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_delete_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_dockings() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_drag() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_drag() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_drag() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
on_drag() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
on_drag() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_draw_background() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
on_draw_graph() (maverig.views.attributePanelView.AttributeCellRenderer method), 50
on_elements() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_elements() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
on_elements() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_error() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
on_exit_presenter_section() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_file_new_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_file_open_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_file_open_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_file_save_as_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_file_save_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_file_save_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_forward_to_end_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_forward_to_end_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_go_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_home_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_help_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_increase_speed_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_increase_speed_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41

on_language() (maverig.presenter.attributePanelPresenter.AttributePanelPresenter method), 31
on_language() (maverig.presenter.consolePanelPresenter.ConsolePanelPresenter method), 33
on_language() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_language() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
on_language() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_language() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 38
on_language() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
on_language() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_language_changed() (maverig.presenter.settingsPresenter.SettingsPresenter method), 39
on_mode() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_mode() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_mode() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
on_mode() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
on_mode() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
on_mouse_released() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_new_simulator_triggered() (maverig.presenter.componentWizardPresenter.ComponentWizardPresenter method), 33
on_output() (maverig.presenter.consolePanelPresenter.ConsolePanelPresenter method), 34
on_param() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_param() (maverig.presenter.group_presenter.lineGroupPresenter.LineGroupPresenter method), 30
on_param() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 38
on_paste_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_pause_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_position_changed() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_position_changed() (maverig.views.items.abstractItem.AbstractItem method), 44
on_position_changed() (maverig.views.items.circle.Circle method), 44
on_position_changed() (maverig.views.items.icon.Icon method), 44
on_position_changed() (maverig.views.items.line.Line method), 45
on_positions() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_program_mode() (maverig.presenter.attributePanelPresenter.AttributePanelPresenter method), 32
on_program_mode() (maverig.presenter.consolePanelPresenter.ConsolePanelPresenter method), 34
on_program_mode() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_program_mode() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 33
on_program_mode() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_program_mode() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 38
on_program_mode() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
on_program_mode() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_progress() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_quit_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter static method), 34
on_raster_mode_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_redo_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_save_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_save_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_run_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_run_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_screen_dateformat() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_select_all_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_selection() (maverig.presenter.attributePanelPresenter.AttributePanelPresenter method), 31
on_selection() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
on_selection() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_selection() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 38
on_selection() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_selection_changed() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter

method), 38
on_selection_mode_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_set_time_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_settings() (maverig.presenter.abstractPresenter.AbstractPresenter method), 31
on_settings_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_settings_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_sim() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_sim() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_slider_moved() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_stop_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_stop_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_trigger_attribute_panel() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_trigger_component_panel() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_trigger_console() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_trigger_progress_bar() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_trigger_property_panel() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_trigger_status_bar() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_undo_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
on_vid_speed() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 36
on_vid_speed() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
on_vid_speed() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 42
on_vid_speed_event() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
on_zoom_fit_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_zoom_fit_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_zoom_in_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_zoom_in_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
on_zoom_out_triggered() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 35
on_zoom_out_triggered() (maverig.presenter.toolbarPresenter.ToolbarPresenter method), 41
opacity (maverig.views.items.abstractItem.AbstractItem attribute), 44
open() (maverig.presenter.menuBarPresenter.MenuBarPresenter method), 34
open_file_dialog() (maverig.views.propertyPanelView.PropertyPanelView method), 53
origin (maverig.views.positioning.vPoint.Change attribute), 47

P

paint_datetime() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
palette_iwanthue50() (in module maverig.utils.colorTools), 56
palette_tableau10() (in module maverig.utils.colorTools), 56
param_is_multivalue() (maverig.models.model.Model method), 24
ParameterTab (class in maverig.views.componentWizardView), 51
parent_item (maverig.views.positioning.vPoint.VPoint attribute), 48
participant() (in module maverig.views.dialogs), 52
paste_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
paste_from_clipboard() (maverig.models.model.Model method), 24
path (maverig.presenter.menuBarPresenter.MenuBarPresenter attribute), 34
popup_triggered (maverig.views.componentWizardView.ComboUpdatingBox attribute), 52
port (maverig.models.model.ElemPort attribute), 19
pos (maverig.views.positioning.vPoint.VPoint attribute), 48
position_changed (maverig.views.positioning.vPoint.VPoint attribute), 47
PresenterManager (class in maverig.presenter.presenterManager), 36
ProcessServer (class in maverig.utils.processServer), 59
program_mode (maverig.models.model.Model attribute), 22
ProgramMode (class in maverig.models.model), 19
ProgressPresenter (class in maverig.presenter.progressPresenter), 37
ProgressView (class in maverig.views.progressView), 53

property_grid (maverig.views.propertyPanelView.PropertyPanelView attribute), 54
property_value_objects (maverig.views.propertyPanelView.PropertyPanelView attribute), 54
PropertyPanelPresenter (class in maverig.presenter.propertyPanelPresenter), 37
PropertyPanelView (class in maverig.views.propertyPanelView), 53

Q

q_color() (in module maverig.utils.colorTools), 55
qdatetime_to_datetime() (maverig.presenter.menuBarPresenter.MenuBarPresenter static method), 36

R

raster_mode (maverig.models.model.Model attribute), 23
raster_pos() (in module maverig.data.config), 18
raster_snap() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
raster_snap_mode (maverig.models.model.Model attribute), 23
raster_snap_v_points (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter attribute), 28
raster_snapped (maverig.views.positioning.vPoint.Change attribute), 47
read_components() (in module maverig.data.config), 16
read_config() (in module maverig.data.config), 17
read_json() (in module maverig.data.config), 16
read_simulators() (in module maverig.data.config), 16
redo() (maverig.models.model.Model method), 23
refreshBg() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
register_function() (maverig.utils.processServer.Server method), 59
register_process_factory() (maverig.utils.processServer.ProcessServer method), 59
remove() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
remove() (maverig.views.groups.abstractGroup.AbstractGroup method), 43
remove() (maverig.views.items.abstractItem.AbstractItem method), 44
remove_item() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
remove_selected_component() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
reset() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
resizeEvent() (maverig.views.attributePanelView.AttributeCell method), 49
restore_config() (in module maverig.data.config), 17
restore_default_components() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
row_has_text() (maverig.utils.tableWidgets.AutoRowTableWidget method), 62
run() (maverig.models.modelSimulation.SimulationProcess method), 26
run() (maverig.utils.processServer.Server method), 59
run() (maverig.views.positioning.section.Section method), 45
run_force_layout() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
run_iteration() (maverig.models.modelSimulation.SimulationServer method), 27
run_iteration() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
run_iteration() (maverig.presenter.utils.forceEngine.ForceEngine method), 31
run_refresh() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
run_slider() (maverig.presenter.progressPresenter.ProgressPresenter method), 37

S

saved_scenario_copy (maverig.models.model.Model attribute), 21
scenario (maverig.models.model.Model attribute), 23
ScenarioBaseVoltageLevelError, 60
ScenarioComponentError, 61
ScenarioConnectionError, 60
ScenarioDatafileError, 61
ScenarioElementError, 61
ScenarioEmptyError, 60
ScenarioError, 59
ScenarioFileNotFoundError, 61
ScenarioOfflineElementError, 60
ScenarioPanelPresenter (class in maverig.presenter.scenarioPanelPresenter), 38
ScenarioPanelView (class in maverig.views.scenarioPanelView), 54
ScenarioRefBusConnectionError, 60
ScenarioRefBusCountError, 60
ScenarioRefBusMissingError, 60
ScenarioSimulationBranchLengthError, 61
ScenarioSimulationPowerflowError, 62
ScenarioSimulationRuntimeError, 62
ScenarioSimulatorError, 61
Section (class in maverig.views.positioning.section), 45
SectionManager (class in maverig.views.positioning.section), 45
select_all_active_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39

select_all_elements() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39
selectable (maverig.views.items.abstractItem.AbstractItem attribute), 44
selected (maverig.views.groups.abstractGroup.AbstractGroup attribute), 42
selected (maverig.views.items.abstractItem.AbstractItem attribute), 44
selection (maverig.models.model.Mode attribute), 20
selection (maverig.models.model.Model attribute), 23
selection_dragging (maverig.models.model.Model attribute), 23
selection_mode_btn_clicked() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
SEPARATOR() (in module maverig.data.config), 18
Server (class in maverig.utils.processServer), 58
ServerProxy (class in maverig.utils.processServer), 59
set_color_effect() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
set_color_effect() (maverig.views.items.abstractItem.AbstractItem method), 44
set_consumer_bar_effect() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
set_consumer_bar_effect() (maverig.views.items.abstractItem.AbstractItem method), 44
set_effect() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
set_graph_visibility() (maverig.views.attributePanelView.AttributeCell method), 49
set_new_accepted_value() (maverig.views.propertyPanelView.PropertyPanelView method), 54
set_param_value() (maverig.models.model.Model method), 24
set_parameter_style() (maverig.views.propertyPanelView.PropertyPanelView method), 54
set_pos() (maverig.models.model.Model method), 25
set_pos() (maverig.models.modelGraph.ModelGraph method), 26
set_pos() (maverig.views.positioning.vPoint.VPoint method), 48
set_producer_bar_effect() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
set_producer_bar_effect() (maverig.views.items.abstractItem.AbstractItem method), 44
set_selected() (maverig.models.model.Model method), 25
set_shadow_effect() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
set_shadow_effect() (maverig.views.items.abstractItem.AbstractItem method), 44
set_state_of_charge_bar() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
set_state_of_charge_bar() (maverig.views.items.abstractItem.AbstractItem method), 44
set_state_of_charge_tip() (maverig.views.groups.abstractGroup.AbstractGroup method), 42
set_state_of_charge_tip() (maverig.views.items.abstractItem.AbstractItem method), 44
set_state_of_charge_tip_bg() (maverig.views.groups.abstractGroup.AbstractGroup method), 43
set_state_of_charge_tip_bg() (maverig.views.items.abstractItem.AbstractItem method), 44
set_svg_icon() (maverig.views.modePanelView.ModePanelView static method), 50
set_svg_icon() (maverig.views.propertyPanelView.PropertyPanelView static method), 54
setGeometry() (maverig.utils.flowlayout.FlowLayout method), 57
Setting (class in maverig.data.settings.abstractSettings), 15
Settings (class in maverig.data.settings.settings), 16
SettingsPresenter (class in maverig.presenter.settingsPresenter), 39
SettingsView (class in maverig.views.settingsView), 54
SettingTypes (class in maverig.data.settings.settings), 16
setUp() (maverig.tests.test_consolePresenter.TestConsolePresenter method), 64
setUp() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 64
setUp() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
setUp() (maverig.tests.test_model.TestModel method), 67
setUp() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
setUp() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
setUp() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
setUp() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
setUp() (maverig.tests.test_settingsPresenter.TestSettingsPresenter method), 71
setUp() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
setUp() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72

setUp() (maverig.tests.test_vPoint.TestVPoint method), 73
short_name() (maverig.models.modelSimulation.SimulationProcess method), 27
show() (maverig.views.dialogs.SimulationTimeDialog method), 52
show() (maverig.views.settingsView.SettingsView method), 54
SHOW_INVISIBLE_COMPONENTS (maverig.data.config.ConfigKeys attribute), 18
show_invisible_components() (maverig.presenter.modePanelPresenter.ModePanelPresenter method), 32
show_license() (in module maverig.views.dialogs), 52
showPopup() (maverig.views.componentWizardView.ComboBoxWithBorder method), 52
sim (maverig.models.model.Mode attribute), 20
sim_end (maverig.models.model.Model attribute), 22
sim_end_index (maverig.models.model.Model attribute), 22
sim_index (maverig.models.model.Model attribute), 22
sim_progress (maverig.models.model.Model attribute), 22
sim_start (maverig.models.model.Model attribute), 22
sim_step_size (maverig.models.model.Model attribute), 22
sim_timestamp (maverig.models.model.Model attribute), 22
simulation (maverig.models.model.Model attribute), 21
simulation (maverig.models.model.ProgramMode attribute), 20
SIMULATION_COMPLETED() (in module maverig.data.config), 18
simulation_paused (maverig.models.model.ProgramMode attribute), 20
SIMULATION_PAUSED() (in module maverig.data.config), 18
SIMULATION_SETTINGS (maverig.data.config.ConfigKeys attribute), 17
simulation_settings_tab (maverig.data.settings.settings.Settings attribute), 16
SIMULATION_SPEED() (in module maverig.data.config), 18
SimulationProcess (class in maverig.models.modelSimulation), 26
SimulationServer (class in maverig.models.modelSimulation), 27
SimulationTimeDialog (class in maverig.views.dialogs), 52
simulator_update() (maverig.views.componentWizardView.IntroPage method), 51
simulators (maverig.models.model.Model attribute), 22
sizeHint() (maverig.utils.flowlayout.FlowLayout method), 57
snap_lock() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
snap_zone() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
snapped (maverig.views.positioning.vPoint.Change attribute), 47
SpeedModel (class in maverig.utils.forceatlas2), 57
SPLITTER_LEFT_GEOMETRY (maverig.data.config.ConfigKeys attribute), 17
SPLITTER_LEFT_STATE (maverig.data.config.ConfigKeys attribute), 17
SPLITTER_MAIN_GEOMETRY (maverig.data.config.ConfigKeys attribute), 17
SPLITTER_MAIN_STATE (maverig.data.config.ConfigKeys attribute), 17
SPLITTER_RIGHT_GEOMETRY (maverig.data.config.ConfigKeys attribute), 17
SPLITTER_RIGHT_STATE (maverig.data.config.ConfigKeys attribute), 17
start() (maverig.models.modelSimulation.SimulationServer method), 27
start() (maverig.utils.processServer.ProcessServer method), 59
start() (maverig.utils.processServer.Server method), 59
start_element() (maverig.models.modelSimulation.SimulationProcess method), 27
start_scoll_to_center_anim() (maverig.views.attributePanelView.AttributeCell method), 49
start_simulation() (maverig.models.modelSimulation.SimulationProcess method), 27
start_simulator() (maverig.models.modelSimulation.SimulationProcess method), 27
state_checked (maverig.views.componentWizardView.AttributeTab attribute), 51
state_checked (maverig.views.componentWizardView.ParameterTab attribute), 51
StatusBarPresenter (class in maverig.presenter.statusBarPresenter), 40
StatusBarView (class in maverig.views.statusBarView), 55
step() (maverig.utils.maverig_csv.CSV method), 58
step() (maverig.utils.visSimulator.VisSimulator method), 63
stop() (maverig.models.modelSimulation.SimulationServer method), 27

stop() (maverig.utils.processServer.ProcessServer method), 59
stop() (maverig.utils.processServer.Server method), 59
stop_simulation() (maverig.models.model.Model method), 26
stop_slider() (maverig.presenter.progressPresenter.ProgressPresenter method), 37
StreamToLogger (class in maverig.utils.logger), 57
success() (maverig.presenter.statusBarPresenter.StatusBarPresenter method), 40
switch_modes() (maverig.models.model.Model method), 23

T

Tab (class in maverig.data.settings.abstractSettings), 15
tabInserted() (maverig.views.componentWizardView.Tabs method), 51
tabRemoved() (maverig.views.componentWizardView.Tabs method), 51
Tabs (class in maverig.views.componentWizardView), 51
tabs (maverig.data.settings.settings.Settings attribute), 16
takeAt() (maverig.utils.flowlayout.FlowLayout method), 56
test_add_history_point() (maverig.tests.test_model.TestModel method), 67
test_adjust_scene_rect() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
test_apply_setting() (maverig.tests.test_settingsPresenter.TestSettingsPresenter method), 71
test_attr_is_multivalue() (maverig.tests.test_model.TestModel method), 69
test_avoid_invalid_positions() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_can_dock() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_can_dock() (maverig.tests.test_model.TestModel method), 68
test_change_ev_icon() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 66
test_changes_count() (maverig.tests.test_model.TestModel method), 68
test_check_snap_permission() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_check_spinbox() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
test_comp_btn_clicked() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_comp_btn_created() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_connectables() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_copy() (maverig.tests.test_model.TestModel method), 68
test_copy_selected_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_copy_to_clipboard() (maverig.tests.test_model.TestModel method), 67
test_create_element() (maverig.tests.test_model.TestModel method), 67
test_cut_selected_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_damped_mouse_pos() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
test_delete_element() (maverig.tests.test_model.TestModel method), 67
test_delete_selected_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_deselect_all_elems() (maverig.tests.test_model.TestModel method), 68
test_dock() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_dock() (maverig.tests.test_model.TestModel method), 68
test_docking_attrs() (maverig.tests.test_model.TestModel method), 69
test_docking_port() (maverig.tests.test_model.TestModel method), 68
test_dockings_in() (maverig.tests.test_model.TestModel method), 68
test_dockings_out() (maverig.tests.test_model.TestModel method), 68
test_drag_started() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 64
test_drop_event() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
test_ep() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 64
test_error() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_fire() (maverig.tests.test_event.TestEvent method), 64
test_fix() (maverig.tests.test_vPoint.TestVPoint method), 77
test_follow_follows() (maverig.tests.test_vPoint.TestVPoint method), 78
test_from_dockables() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_get_attr_value() (maverig.tests.test_model.TestModel method), 69

test_get_attr_values() (maverig.tests.test_model.TestModel method), 68
test_get_i_heat_value() (maverig.tests.test_model.TestModel method), 69
test_get_icon_color() (maverig.tests.test_model.TestModel method), 68
test_get_p_level() (maverig.tests.test_model.TestModel method), 69
test_get_param_value() (maverig.tests.test_model.TestModel method), 68
test_get_pos() (maverig.tests.test_model.TestModel method), 68
test_get_selected() (maverig.tests.test_model.TestModel method), 68
test_get_shared_published_attrs() (maverig.tests.test_model.TestModel method), 69
test_get_shared_published_params() (maverig.tests.test_model.TestModel method), 67
test_get_simulator() (maverig.tests.test_model.TestModel method), 68
test_get_state_of_charge() (maverig.tests.test_model.TestModel method), 69
test_get_u_heat_value() (maverig.tests.test_model.TestModel method), 69
test_group_presenters() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
test_groups() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
test_hand_mode_btn_clicked() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_handle() (maverig.tests.test_event.TestEvent method), 64
test_handle_scenario_error() (maverig.tests.test_model.TestModel method), 69
test_hide_selected_component() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_info() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_init_history() (maverig.tests.test_model.TestModel method), 67
test_init_scene_mapping() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_install_language() (maverig.tests.test_settingsPresenter.TestSettingsPresenter method), 71
test_is_selectable() (maverig.tests.test_model.TestModel method), 67
test_mouse_clicked() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 70
test_mouse_moved() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_mouse_released() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_move_pos() (maverig.tests.test_vPoint.TestVPoint method), 73
test_new_connectables() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_non_connectable() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_on_back_to_start_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_change_dateformat() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_change_visibility_triggered() (maverig.tests.test_consolePresenter.TestConsolePresenter method), 64
test_on_change_visibility_triggered() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 64
test_on_change_visibility_triggered() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_change_visibility_triggered() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
test_on_change_visibility_triggered() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_on_clipboard() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_console_clear_triggered() (maverig.tests.test_consolePresenter.TestConsolePresenter method), 64
test_on_dockings() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 66
test_on_drag() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 66
test_on_drag() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_drag() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72
test_on_elements() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_on_elements() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_on_elements() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72
test_on_scenario_panel_triggered()

(maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_go_to_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_hand_mode_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_increase_speed_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_mode() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 66
test_on_mode() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_mode() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 64
test_on_mode() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_on_mode() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_on_mouse_released() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_on_output() (maverig.tests.test_consolePresenter.TestConsolePresenter method), 64
test_on_param() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 66
test_on_param() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
test_on_pause_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_position_changed() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_on_positions() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_on_program_mode() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_program_mode() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_program_mode() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
test_on_program_mode() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_on_program_mode() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72
test_on_progress() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_raster_mode_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_reduce_speed_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_run_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_screen_dateformat() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_selection() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 66
test_on_selection() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_selection() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
test_on_selection() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72
test_on_selection_mode_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_set_time_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_speed() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_speed() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72
test_on_slider_moved() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 69
test_on_stop_triggered() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 66
test_on_vid_speed() (maverig.tests.test_menuBarPresenter.TestMenusBarPresenter method), 67
test_on_vid_speed() (maverig.tests.test_toolbarPresenter.TestToolBarPresenter method), 72
test_on_vid_speed_event() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_param_is_multivalue() (maverig.tests.test_model.TestModel method), 69
test_paste_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_paste_from_clipboard() (maverig.tests.test_model.TestModel method), 67
test_program_mode() (maverig.tests.test_consolePresenter.TestConsolePresenter method), 64
test_program_mode() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 64
test_progress_reset() (maverig.tests.test_groupPresenter.TestGroupPresenter

method), 65
test_raster_snap_v_points() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 64
test_redo() (maverig.tests.test_model.TestModel method), 67
test_remove() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_remove_selected_component_and_restore_default_components() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_reset() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_run_force_layout() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_run_iteration() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 70
test_run_slider() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 70
test_select_all_active_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_select_all_elements() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_selection_mode_btn_clicked() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_set_param_value() (maverig.tests.test_model.TestModel method), 68
test_set_pos() (maverig.tests.test_model.TestModel method), 68
test_set_pos() (maverig.tests.test_vPoint.TestVPoint method), 73
test_set_selected() (maverig.tests.test_model.TestModel method), 68
test_show_invisible_components() (maverig.tests.test_modePanelPresenter.TestModePanelPresenter method), 63
test_snap_dock() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_snap_zone() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_stop_slider() (maverig.tests.test_progressPresenter.TestProgressPresenter method), 70
test_success() (maverig.tests.test_statusBarPresenter.TestStatusBarPresenter method), 72
test_switch_modes() (maverig.tests.test_model.TestModel method), 67
test_to_dockables() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_undo() (maverig.tests.test_model.TestModel method), 67
test_undock() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
test_undock() (maverig.tests.test_model.TestModel method), 68
test_unfix() (maverig.tests.test_vPoint.TestVPoint method), 73
test_unfollow() (maverig.tests.test_vPoint.TestVPoint method), 73
test_unhandle() (maverig.tests.test_event.TestEvent method), 64
test_update() (maverig.tests.test_model.TestModel method), 68
test_validate_all() (maverig.tests.test_model.TestModel method), 68
test_validate_scenario() (maverig.tests.test_model.TestModel method), 69
test_zoom() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 65
text_changed() (maverig.tests.test_propertyPanelPresenter.TestPropertyPanelPresenter method), 70
test_vp() (maverig.tests.test_groupPresenter.TestGroupPresenter method), 64
test_zoom() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
test_zoom() (maverig.tests.test_scenarioPanelPresenter.TestScenarioPanelPresenter method), 71
TestConsolePresenter (class in maverig.tests.test_consolePresenter), 64
TestEvent (class in maverig.tests.test_event), 64
TestGroupPresenter (class in maverig.tests.test_groupPresenter), 64
TestMenusBarPresenter (class in maverig.tests.test_menuBarPresenter), 66
TestModel (class in maverig.tests.test_model), 67
TestModePanelPresenter (class in maverig.tests.test_modePanelPresenter), 63
TestProgressPresenter (class in maverig.tests.test_progressPresenter), 69
TestPropertyPanelPresenter (class in maverig.tests.test_propertyPanelPresenter), 70
TestScenarioPanelPresenter (class in maverig.tests.test_scenarioPanelPresenter), 70
TestSettingsPresenter (class in maverig.tests.test_settingsPresenter), 71
TestStatusBarPresenter (class in maverig.tests.test_statusBarPresenter), 72
TestToolBarPresenter (class in maverig.tests.test_toolbarPresenter), 72
TestVPoint (class in maverig.tests.test_vPoint), 73
text (maverig.utils.scenarioErrors.ScenarioBaseVoltageLevelError attribute), 60
text (maverig.utils.scenarioErrors.ScenarioComponentError

attribute), 61
text (maverig.utils.scenarioErrors.ScenarioConnectionError attribute), 60
text (maverig.utils.scenarioErrors.ScenarioDatafileError attribute), 61
text (maverig.utils.scenarioErrors.ScenarioElementError attribute), 61
text (maverig.utils.scenarioErrors.ScenarioEmptyError attribute), 60
text (maverig.utils.scenarioErrors.ScenarioError attribute), 59
text (maverig.utils.scenarioErrors.ScenarioFileNotFoundError attribute), 61
text (maverig.utils.scenarioErrors.ScenarioOfflineElementError attribute), 60
text (maverig.utils.scenarioErrors.ScenarioRefBusConnectionError attribute), 60
text (maverig.utils.scenarioErrors.ScenarioRefBusCountError attribute), 61
text (maverig.utils.scenarioErrors.ScenarioRefBusMissingError attribute), 60
text (maverig.utils.scenarioErrors.ScenarioSimulationBranchLengthError attribute), 62
text (maverig.utils.scenarioErrors.ScenarioSimulationPowerflowError attribute), 62
text (maverig.utils.scenarioErrors.ScenarioSimulationRuntimeError attribute), 62
text (maverig.utils.scenarioErrors.ScenarioSimulatorError attribute), 61
text() (maverig.utils.tableWidgets.AutoRowTableWidget method), 62
text_changed() (maverig.utils.tableWidgets.CellLineEdit method), 62
title (maverig.utils.scenarioErrors.ScenarioBaseVoltageLevelError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioComponentError attribute), 61
title (maverig.utils.scenarioErrors.ScenarioConnectionError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioDatafileError attribute), 61
title (maverig.utils.scenarioErrors.ScenarioElementError attribute), 61
title (maverig.utils.scenarioErrors.ScenarioEmptyError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioError attribute), 59
title (maverig.utils.scenarioErrors.ScenarioFileNotFoundError attribute), 61
title (maverig.utils.scenarioErrors.ScenarioOfflineElementError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioRefBusConnectionError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioRefBusCountError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioRefBusMissingError attribute), 60
title (maverig.utils.scenarioErrors.ScenarioSimulationBranchLengthError attribute), 61
title (maverig.utils.scenarioErrors.ScenarioSimulationPowerflowError attribute), 62
title (maverig.utils.scenarioErrors.ScenarioSimulationRuntimeError attribute), 62
title (maverig.utils.scenarioErrors.ScenarioSimulatorError attribute), 61
tmp_scenario_copy (maverig.models.model.Model attribute), 21
tmp_undockables() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
ToolbarPresenter (class in maverig.presenter.toolbarPresenter), 40
ToolbarView (class in maverig.views.toolbarView), 55
translate() (maverig.views.attributePanelView.AttributeCell method), 49
translate() (maverig.views.attributePanelView.AttributePanelView method), 49
translate() (maverig.views.consolePanelView.ConsolePanelView method), 52
translate() (maverig.views.progressView.ProgressView method), 53
trigger_section (maverig.views.positioning.vPoint.VPoint attribute), 47

U

UI_STATE (maverig.data.config.ConfigKeys attribute), 17
uid (maverig.models.model.Model attribute), 21
uncheck_buttons() (maverig.views.modePanelView.ModePanelView method), 50
undo() (maverig.models.model.Model method), 23
undock() (maverig.models.model.Model method), 25
undock() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
unfix() (maverig.views.positioning.vPoint.VPoint method), 48
unfollow() (maverig.views.positioning.vPoint.VPoint method), 48
unhandle() (maverig.utils.event.Event method), 56
unhover_component_button() (maverig.views.modePanelView.ModePanelView method), 50
update() (maverig.models.model.Model method), 26
update_all() (maverig.models.model.Model method), 26
update_data() (maverig.models.modelSimulation.SimulationServer method), 27
update_graph_visibility() (maverig.presenter.attributePanelPresenter.AttributePanelPresenter method), 31

update_info_label() (maverig.views.attributePanelView.AttributePanelView method), 49

V

validate_scenario() (maverig.models.model.Model method), 26
validation() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 29
value_changed() (maverig.presenter.propertyPanelPresenter.PropertyPanelPresenter method), 38
vid_speed (maverig.models.model.Model attribute), 22
vid_speed_rel (maverig.models.model.Model attribute), 22
visible (maverig.views.items.abstractItem.AbstractItem attribute), 44
VisSimulator (class in maverig.utils.visSimulator), 62
vp() (maverig.presenter.group_presenter.abstractGroupPresenter.AbstractGroupPresenter method), 28
VPMouse (class in maverig.views.positioning.vPoint), 48
VPoint (class in maverig.views.positioning.vPoint), 47

W

wheelEvent() (maverig.views.scenarioPanelView.ScenarioPanelView method), 54
write() (maverig.models.modelSimulation.SimulationServer method), 27
write() (maverig.utils.logger.StreamToLogger method), 58
write_config() (in module maverig.data.config), 17
write_json() (in module maverig.data.config), 17

Z

z_value (maverig.views.items.abstractItem.AbstractItem attribute), 43
ZERO_KM_LENGTH() (in module maverig.data.config), 18
zoom() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 38
zoom_fit() (maverig.presenter.scenarioPanelPresenter.ScenarioPanelPresenter method), 39