Technische Universität Berlin

Faculty of Electrical Engineering and Computer Science

Institute of Software Engineering and Theoretical Computer Science

PySimLib - Python Simulation Library

User Guide

Release date: October 16, 2016

This is a work in progress version

Contents

List of Tables

List of Figures

Listings

1 Introduction

This user guide is aimed to aid the user in setting up the Python Simulation Library (in the following abbreviated with PySimLib), run it, and explain the libraries features. After reading this document the user should be able to simulate models using the PySimLib and be able to use it in Python in order to create own simulation descriptions.
If you have any questions or issues, please feel free to write an e-mail to *a.mehlhase@tu-berlin.de*. In case you find bugs in our software we would be pleased for a report at our repository `https://gitlab.tubit.tu-berlin.de/a.mehlhase/PySimulationLibrary`.

2 Setting up PySimLib

The following will show you how to set up PySimLib, while mainly focusing on Windows platforms. Installation instructions for systems based on Ubuntu are also given. Due to the variety of Linux systems we can't provide installation instructions for all of them. Unfortunately we currently have no access to a Mac computer and can therefore not provide installation instructions, nor do we know if PySimLib works on Mac at all. We would be pleased to hear feedback on that.

2.1 Note on versions

PySimLib was developed for Python 3, in particular version 3.5. The library might also work with previous versions of Python 3 but surely not with Python 2. By us, PySimLib has only been tested in Python 3.5.

PySimLib currently supports the simulators listed in table 1. It should be noted that these simulators exist in different versions, which differentiate in terms of features etc. We have listed the constellations of operating system and simulator version under which we tested PySimLib in table 2.

Dymola
OpenModelica
MATLAB/Simulink

Table 1: List of simulators supported by PySimLib

	Windows 7 64 bit	Windows 10 64 bit	Ubuntu 14.10	Ubuntu 16.04
Dymola 2013	✓			
OpenModelica 1.9.6			✓	✓
MATLAB/Simulink R2013a				✓

Table 2: Testing Constellations of Operating Systems and Simulators using PySimLib

2.2 Command line

TODO: wie man ne command line öffnet

2.3 Installing Python

2.3.1 Windows

TODO: bild mit python was angeklickt sein muss

2.3.2 Ubuntu

Python should already be installed on Ubuntu. If not, install it by running the following commands:

```
sudo apt-get install python3
sudo apt-get install python3-pip
```

If you are unsure whether Python is installed or not, run the commands anyways. In case Python is already installed, apt-get is going to inform you and nothing is going to be done.

2.4 Installing a Simulator

In order to simulate models using the PySimLib you will need a simulator that can simulate your models. Note that PySimLib is not a simulator itself but can communicate with several ones and provides a common interface to do so. Please follow the manual of the simulator of your choice in order to set it up correctly. Be sure that it is working properly before continuing with PySimLib. Again, the supported simulators are mentioned in Table **??**.

You can install (or remove) additional simulators any time but whenever you do that, you have to reconfigure PySimLib. See subsection 2.6 for configuring PySimLib.

2.5 Installing PySimLib

PySimLib can be installed very simply using pip. In order to determine the correct pip version try entering the following commands:

```
pip3.5
pip3
pip
```

Remember the first one that worked and take it for all following commands where "pipxyz" is used.
To install PySimLib execute the following commands (**you might need advanced privileges**):

```
pipxyz install zmq
pipxyz install PySimLib
```

2.6 Configuring PySimLib

TODO: schreiben

TODO: weiter