

Simso web - Developer Documentation

Josué Alvarez

July 8, 2015

1 Preamble

This document is a non-exhaustive developer documentation, whose goal is to help future maintainers of the simso-web application to get started with the code. This document is written with the assumption that the developer is using an UNIX environment.

2 Getting Started

Introduction Simso web is a graphical interface built on top of simso. It runs as a full-client application (no server-side) written in javascript, and uses PypyJS (a javascript implementation of Pypy) to run Python in order to execute simso. The main frameworks/tools used for the front-end development are Angular JS and Bootstrap.

2.1 Getting the code

The code is available on github here : <https://github.com/MaximeCheramy/simso-web>. As simso-web embeds its own version of simso, it is included in the form of a git submodule.

To setup your working copy, you have to run the following commands :

```
git clone https://github.com/MaximeCheramy/simso-web.git
cd simso-web/submodules/simso
git submodule init
git submodule update
```

2.2 Architecture

Overview The project is composed of 3 major components :

- The HTML / Javascript / AngularUI front-end.
- Simso : the component used to run the simulation.
- The python bridge between Simso and the Javascript application.

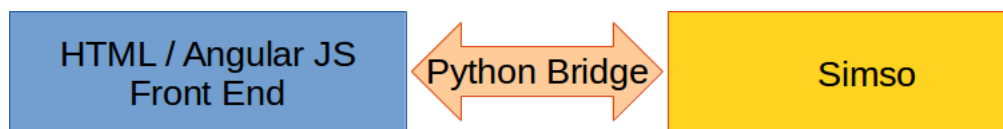


Figure 1: Simso web's major components

Use case In order to understand the structure of the code, we are going to introduce a typical simso-web use case :

1. The user sets up a configuration on the configuration view.

2. The user clicks on the "Run" button
 - A python configuration script is created using the parameters in the configuration view (file js/controllers/config-controller.py). This script is then executed and creates a python 'Configuration' object.
 - The 'run()' method of the Bridge is called (py/simso-bridge.py) and launches the Simso simulation.
 - Once this is done, a variable containing the results of the simulation is created in a way it can be read by the javascript application.
3. The user clicks the "Results" button : the results are now displayed. Some python functions (in py/simso-bridge.py) are used to aggregate results.

The HTML / Javascript / Angular UI front-end The front end is built upon the MVC architecture.

- The Model : the model files are contained within the js/services/ directory. They are called 'services'. The main services are :
 - conf-service.js : contains all the data related to the Configuration of the simulation. That data is going to be used to generate the configuration script.
 - pypy-service.js : contains all the code related to the initialisation of the virtual machine.
- The Views : the view files are contained in the partial/ folder. They are HTML files. There are 2 main views :
 - The Configuration view is the place where the user can setup the simulation parameters.
 - The Results view is the place where the user can see the simulation's results once it has been run.
- The Controllers : the controllers are located within the js/controllers/ directory.

The bridge The bridge is responsible for passing configuration to simso, gathering results and passing them back to javascript.

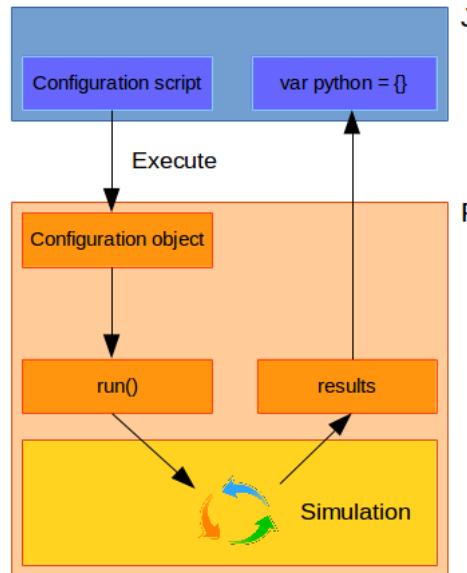


Figure 2: The python bridge

Simso Simso is integrated in simso-web as a submodule. Simso's documentation is available here : <http://homepages.laas.fr/mcheramy/simso/doc>.