

DEVSIMPy-mob : une application mobile hybride pour la simulation à événement discrets des systèmes ubiquitaires

DEVSIMPy-mob: a hybrid mobile app dedicated to the discrete-event simulation of ubiquitous systems

L. Capocchi, S. Sehili, C. Kessler, J.F. Santucci
University of Corsica - SPE UMR CNRS 6134
capocchi@univ-corse.fr



Les journées DEVS francophones : Théorie et application
Institut d'études scientifique de Cargèse
11-15 Avril 2016



Outline

1. Introduction & Context
2. Objectives
3. DEVSimPy and DEVSimPy REST server
4. DEVSimPy-mob mobile app
5. Case study: Phidgets interaction
6. Summary

Outline

1. Introduction & Context
2. Objectives
3. DEVSimPy and DEVSimPy REST server
4. DEVSimPy-mob mobile app
5. Case study: Phidgets interaction
6. Summary

Introduction

- In ubiquitous computing almost any device (connected object) can be embedded with chips to connect the device to an infinite network of other devices (using web services)
- Ubiquitous systems involve complex systems that require M&S-based formalism (like DEVS) and software in order to interact with connected objects using simulation
- DEVSImPy-mob offers a solution to simulate DEVS complex systems (including ubiquitous systems) in a remote way with web services

Context

- The proposed approach strongly associate simulations mobile apps and connected objects
- **DEVSImPy-mob** is a multi-plateform mobile application aimed to manage:
 - discrete event simulations obtained from DEVS (Discrete Event system Specification) models associated with
 - connected objects such as board computers, sensors, controllers or actuators

Outline

1. Introduction & Context
- 2. Objectives**
3. DEVSimPy and DEVSimPy REST server
4. DEVSimPy-mob mobile app
5. Case study: Phidgets interaction
6. Summary

Objectives

- **Goal of DEVSimPy-mob:**
 - to offer DEVSimPy simulation models in mobile devices
 - to interact with DEVS models (connected objects) in order to make simulation in a real and meaningful context: live input model's data, model selection or configuration depending on a context, etc...
 - to simulate the DEVSimPy models in a remote way
 - to improve live decision making on the ground during or after simulation

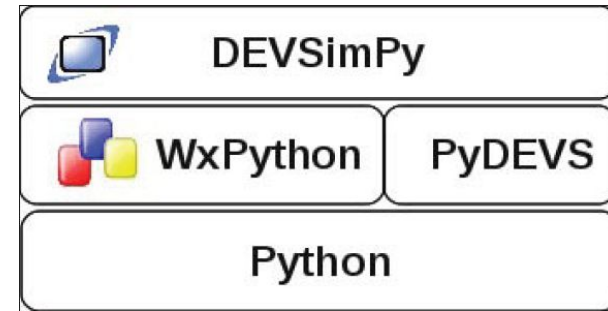
Outline

1. Introduction & Context
2. Objectives
3. **DEVSIMPy and DEVSIMPy REST server**
4. DEVSIMPy-mob mobile app
5. Case study: Phidgets interaction
6. Summary

DEVSimPy

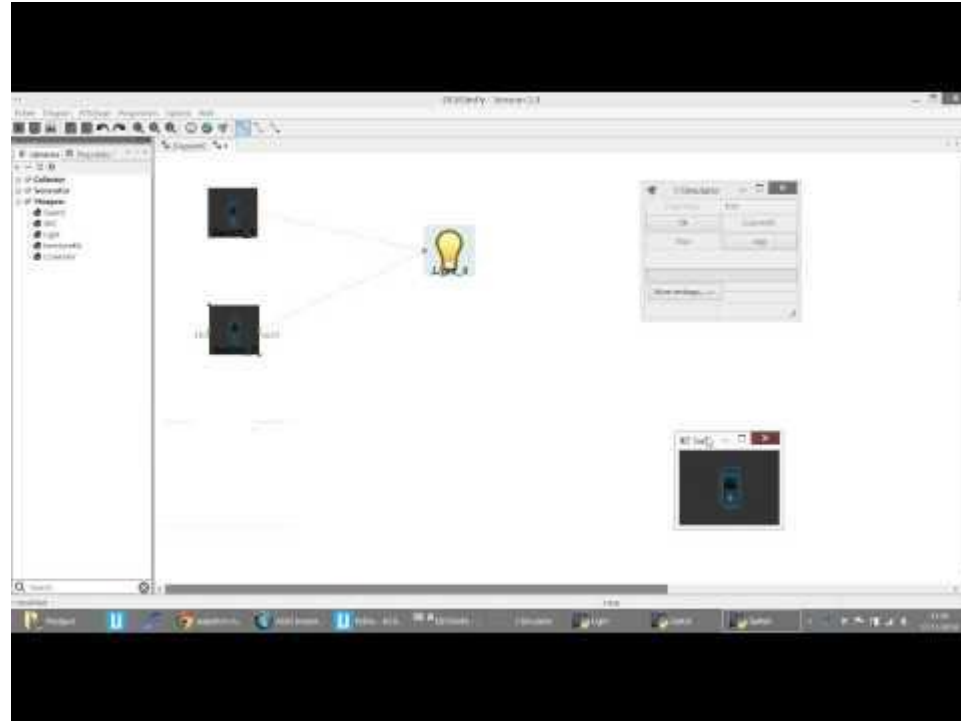
Specification

- Open Source GPL (<https://github.com/capocchi/DEVSimPy>)
- based on Py(P)DEVS simulation kernel (Mc Gill)
- implemented in Python with the MVC approach (GUI is independante of kernels)
- extendable by using plugins
- modeling using model libraries
- a no GUI version of DEVSimPy is available in order to simulate models:
 - *python devsimpy-nogui test.dsp 10*
to simulate the test model during 10 steps.



DEVSimPy

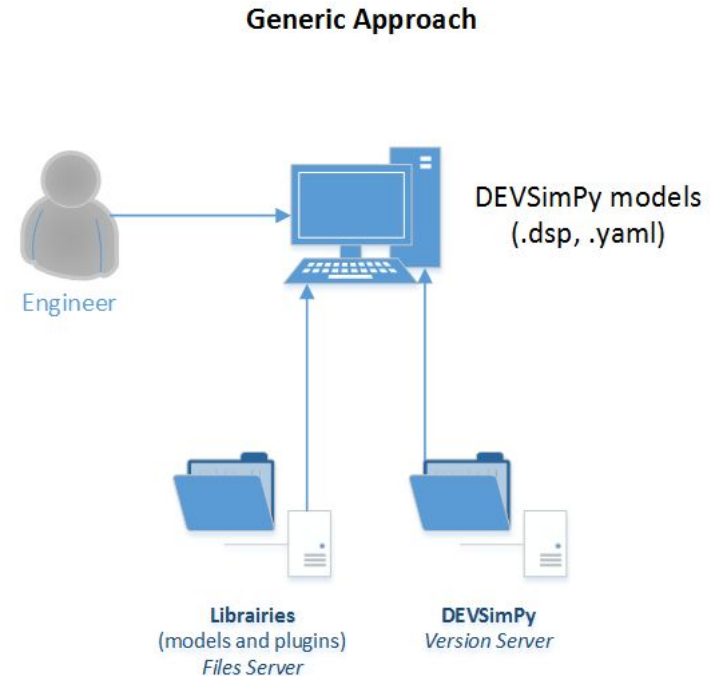
User Interface



DEVSimPy

Methodologie

- Engineers implement their models (.dsp or .yaml) into DEVSimPy using librairies
- The source code of DEVSimPy and libraries are independent and can be managed by separate file servers (Git for source code exemple and DropBox for model librairies for exemple)



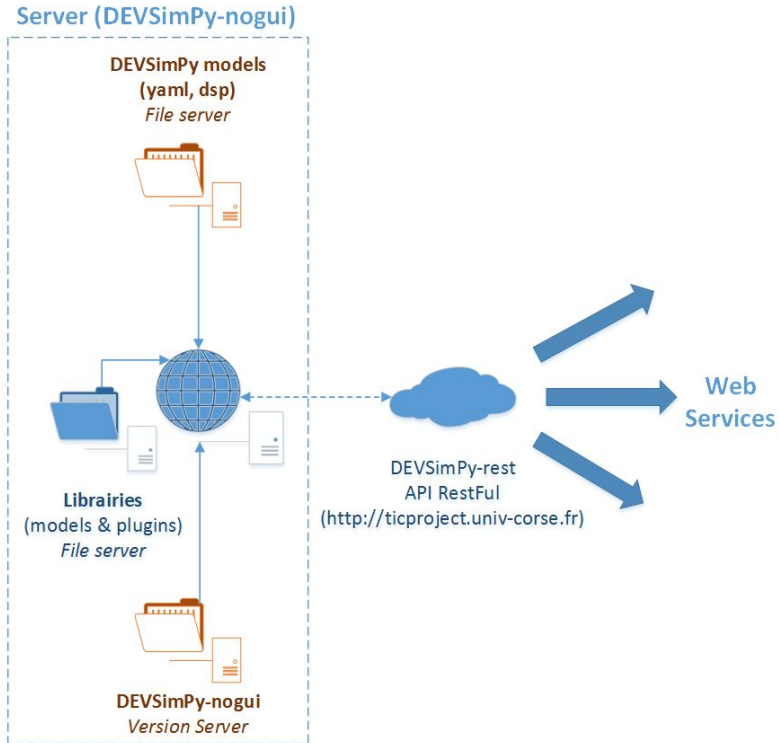
DEVSimPy REST Server

Specifications

- a Restful API web services for DEVSimPy
- Open Source GPL v3.0 (https://github.com/capocchi/DEVSimPy_rest)
- DEVS simulation service through a rest full request
- based on DEVSimPy-nogui
- used **Bottle** - a fast, simple and lightweight WSGI micro web-framework for Python
- The models (.dsp or .yaml) developed with DEVSimPy can be hosted by DEVSimPy-rest server

DEVSimPy REST Server

Architecture



<p>POST /simulation with Content-Type:application/json and body {model_name : XXX simulated_duration : XXX} response : simulation unique id = model_name+date+pid</p>	<p>to simulate mod. yaml during simulated_duration</p>
<p>PUT /simulation/simulation_id/pause</p>	<p>suspend simulation</p>
<p>PUT /simulation/simulation_id/resume</p>	<p>resume simulation</p>
<p>PUT /simulation/simulation_id/kill</p>	<p>kill simulation</p>

Outline

1. Introduction & Context
2. Objectives
3. DEVSimPy and DEVSimPy REST server
- 4. DEVSimPy-mob mobile app**
5. Case study: Phidgets interaction
6. Summary

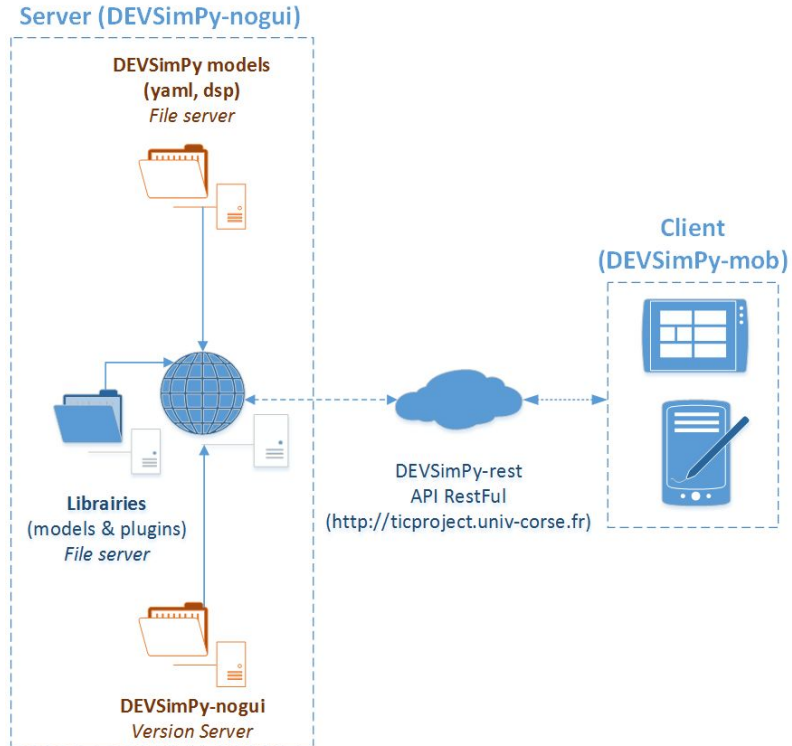
DEVSimPy-mob

Specifications

- hybrid mobile application allowing the simulation of DEVSimPy models from a mobile phone
- Open Source GPL v3.0 (https://github.com/capocchi/DEVSimPy_mob)
- The idea is to allow users of DEVSimPy simulate their models from a mobile phone
- coupled with a [DEVSimPy-rest](#) server which is in charge to simulate DEVSimPy models

DEVSimPy-mob

Architecture



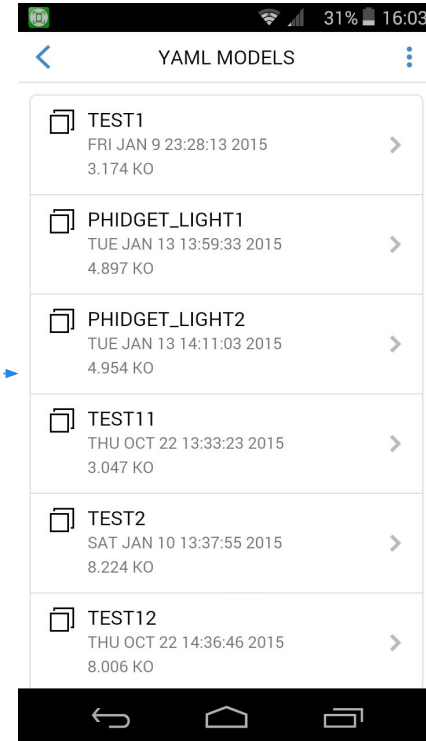
DEVSimPy-mob

Technologies

- Hybrid mobile app based on:
 - [Apache Cordova](#) - open-source mobile development framework.
 - [jQuery-mobile](#) - HTML5-based user interface system designed to make responsive web sites and apps that are accessible on all smartphone, tablet and desktop devices.
 - [Ratchet](#) - front-end framework for building mobile web apps in HTML, CSS, and JavaScript.
 - [FusionCharts](#) - JavaScript Graphics library built on top of raphaeljs.
 - [Joint.js](#) - a diagramming library that focuses on rendering and interacting with diagrams.

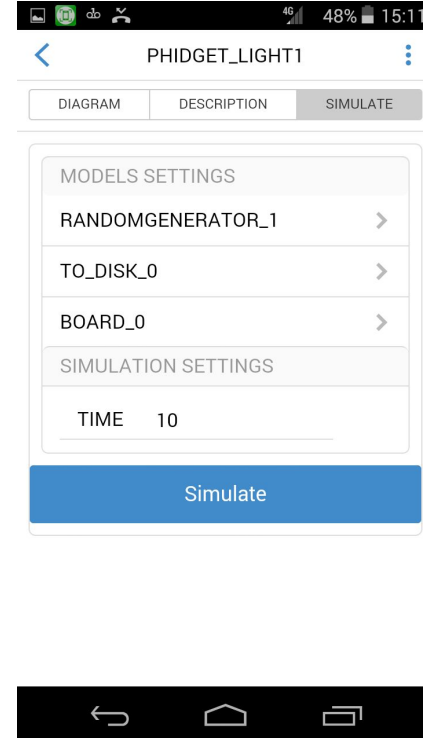
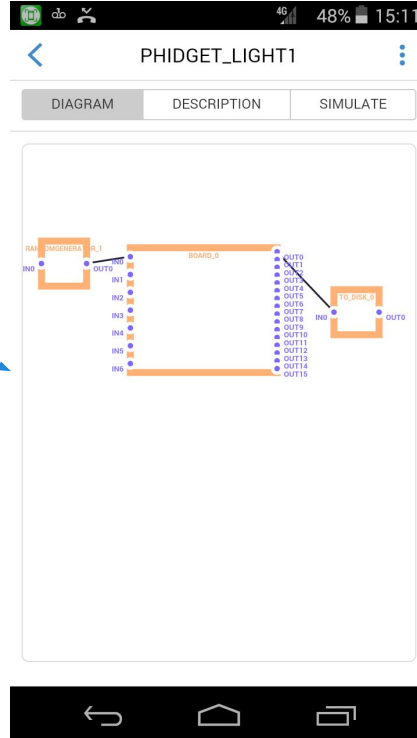
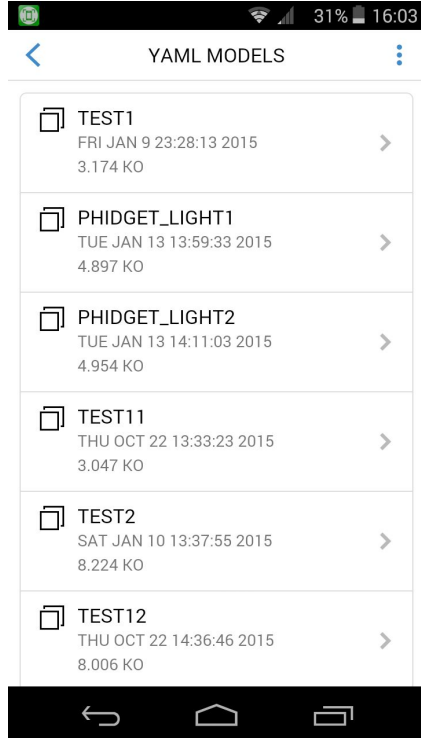
DEVSimPy-mob

User Interface (1)



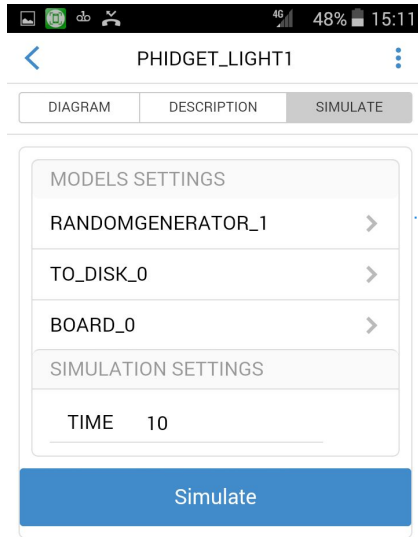
DEVSimPy-mob

User Interface (2)



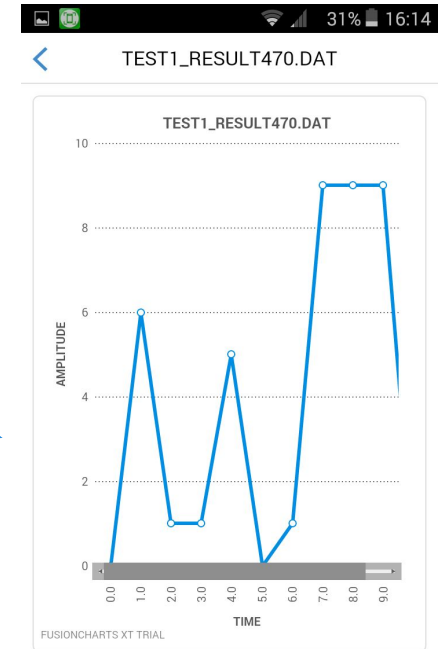
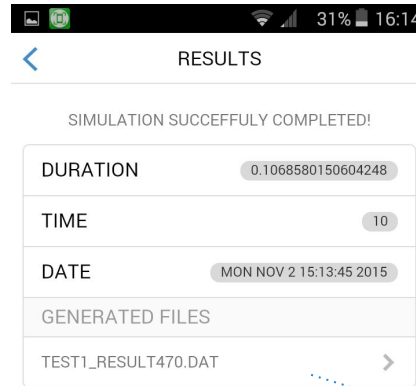
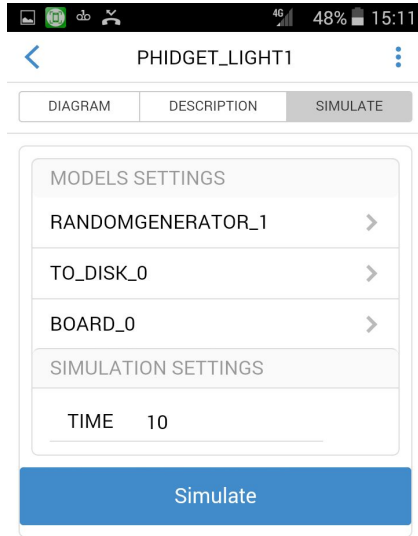
DEVSimPy-mob

User Interface (3)



DEVSimPy-mob

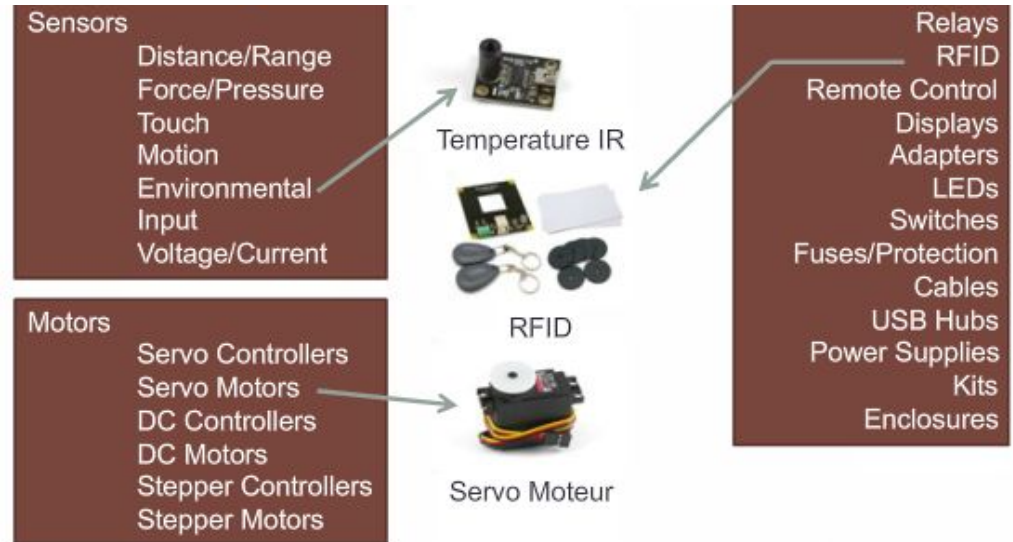
User Interface (4)



Case Study

Phidgets Interaction (1)

Phidget Single Board Computer (SBC)



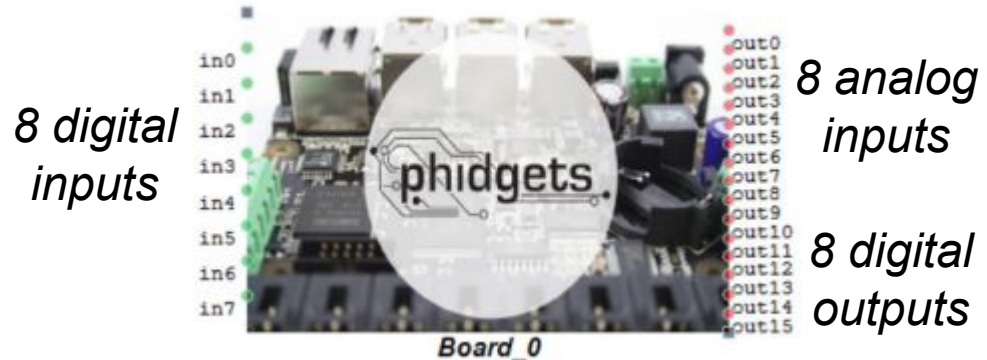
<http://www.phidgets.com/products.php?category=1>

Case Study

Phidgets Interaction (2)

Phidget SBC DEVSimPy modeling:

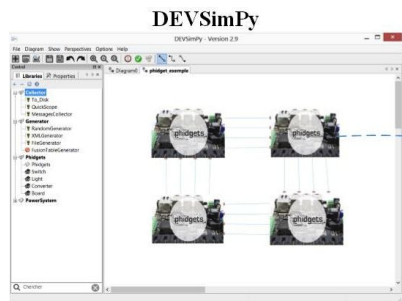
- Phidgets DEVSimPy library offers atomic models based on a Python Phidget API that allow to connect and interact (get sensor values or command actuators) with the Phidget component



Case Study

Phidgets Interaction (3)

Phidget SBC DEVS modeling

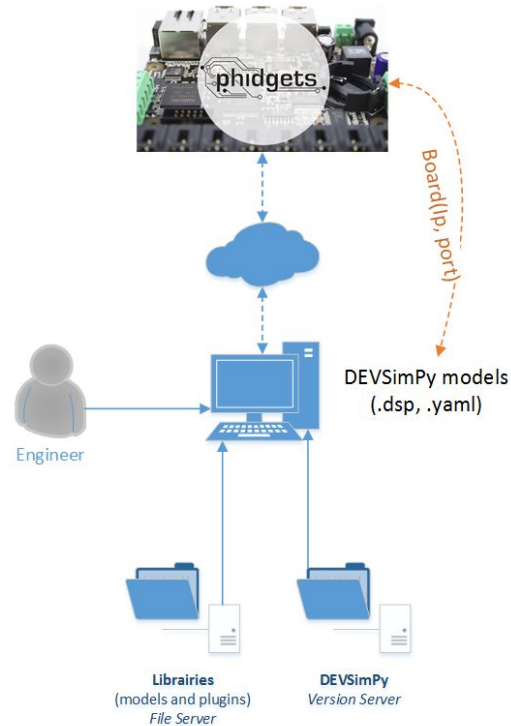
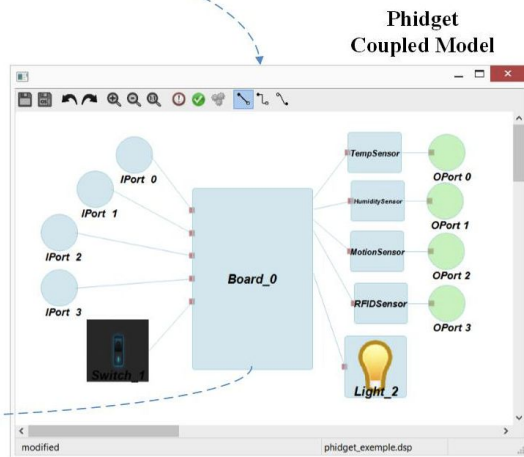


Board_0

Attribute	Value	Information
label	Board_0	Name
label_pos	center	Label position
pen	[#add8e6, 1, 100]	Color and size
fill	[#add8e6]	Background
font	[12, 74, 93, 92, u'Ar-Font label	Background label
image_path		Background
input	7 Input port	
output	16 Output port	
freq	0.200000 sample	
ip	193.48.30.228	IP of the
port	9090	port of the
python_path	Atomic_Name.py	Python file

Doc
DEVS Class for Atomic_Name model
Note: if ip and port is "" the Board is InterfaceKit

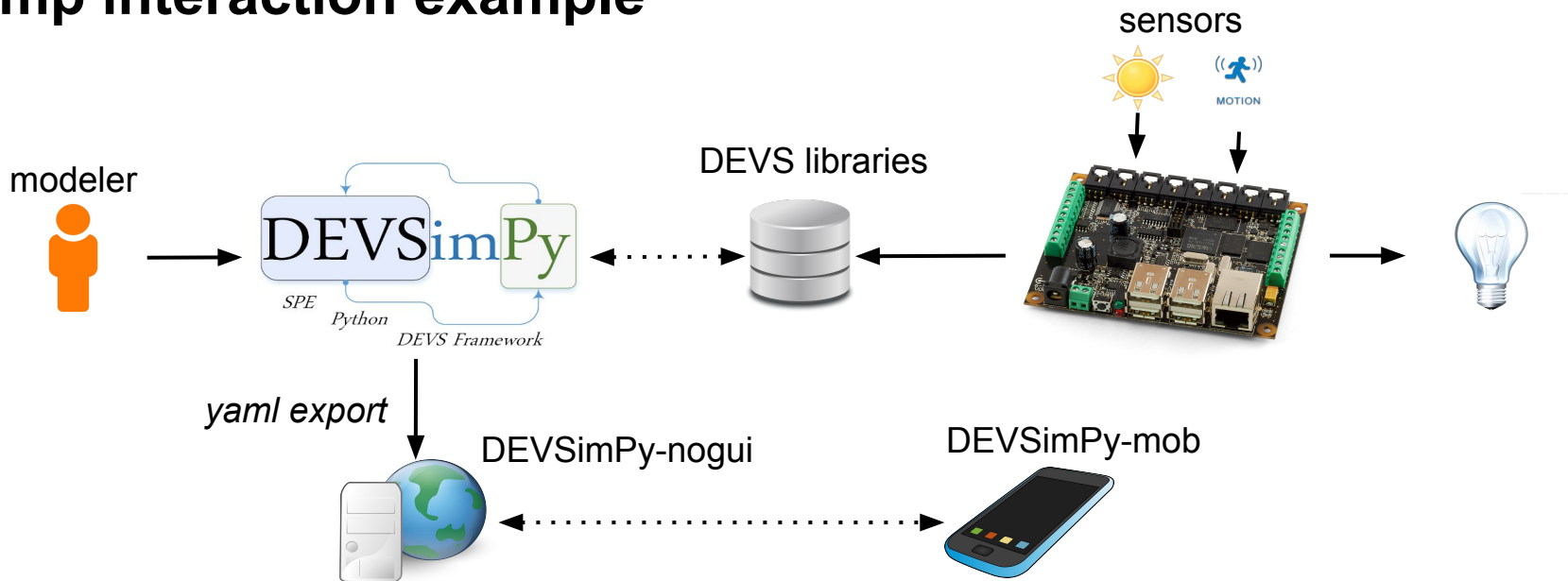
Properties



Case Study

Phidgets Interaction (4)

Lamp interaction example



Outline

1. Introduction & Context
2. Objectives
3. DEVSimPy and DEVSimPy REST server
4. DEVSimPy-mob mobile app
5. Case study: Phidgets interaction
- 6. Summary**

Summary

- **Capabilities of DEVSimPy-Mob:**
 - Multi-Platform (IOS, Android, etc.)
 - Access to the Web Server Services
 - Launch simulations providing intelligent decisions
 - Graphically visualization of DEVS models
- **Future work:**
 - Improve the visualization of the results
 - Interaction with models during simulation
 - Extend the application to health care diagnosis
 - Improve the connexion between DEVSimpy & DEVSimPy-mob