Introduction to Robotframework

Kenneth Bellock

June 9, 2018

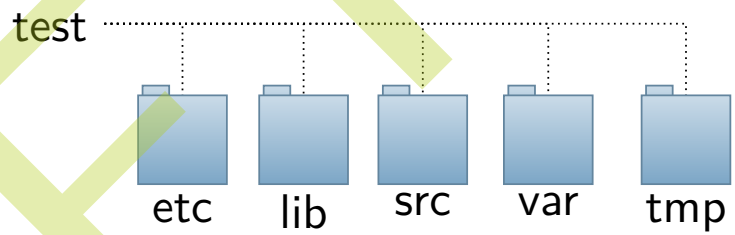
http://robotframework.org

Table of contents

Objectives

Directory Organization

- There is no *"Way of the Robot"* when it comes to organizing your project
- The framework is designed to be flexible, so you can define any organization you want, but this causes a lot of trouble for beginners and often results in projects where everything is just dumped into one big pot
- For these examples, we will use the Filesystem Hierarchy Standard (`https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard`)



etc	Resource files
lib	Libraries
src	Test Suites
var	Variables
tmp	Temporary Files

What is Robotframework?

From their webpage:

Robot Framework is a generic test automation framework for acceptance testing and acceptance test-driven development (ATDD). It has easy-to-use tabular test data syntax and it utilizes the keyword-driven testing approach. Its testing capabilities can be extended by test libraries implemented either with Python or Java, and users can create new higher-level keywords from existing ones using the same syntax that is used for creating test cases.

Installation

- Virtualenv (`https://virualenv.pypa.io`), is a tool to create isolated Python environments; as demonstrated in Listing 1, it will provide the capability to install python toolboxes without administrator priveledges
- For these examples, I will use the locally qualified path to a virtual environment we are going to create
- If you already have an installation of robotframework available, substitute the path to your installation

Listing 1: Install Robotframework

```
1 >> virtualenv local
2 >> local/bin/pip install robotframework
```