Using Fabric

Inland Empire Python User Group
June 17th, 2014

Cameron Goodale
@sigep311

What is Fabric?

Command Line Interface to Python Functions

Task Execution Runner

It also works via SSH

Python 2.5 - 2.7

http://fabfile.org/

http://docs.fabfile.org/en/latest/

How does it work?

Gives you a 'fab' command

Parses fabfile.py in current working dir

Makes functions available

When / Why should I use Fabric?

Use Case 1:

    $ ssh server1

    $ command$_1$, command$_2$, ... command$_n$

    $ exit

    $ repeat...

When / Why should I use Fabric?

Use Case 1 Solution:

    $ fab restart_services

When / Why should I use Fabric?

Use Case 2 (Local Commands):

    Monday

    $ $command_1$, $command_2$, ... $command_n$

    Tuesday

    $ $command_1$, $command_2$, ... $command_n$

    Wednesday

    $ repeat

When / Why should I use Fabric?

Use Case 2 Solution (Local Commands):
Monday
$ fab clean_and_build
Tuesday
$ fab clean_and_build
Wednesday
$ repeat

Demo Code - Setup

$ git clone https://github.com/cgoodale/IEPUG_Fabric.git

$ cd IEPUG_Fabric

$ virtualenv env

$ source env/bin/activate

$ pip install fabric

Demo Code - Using 'fab'

$ fab --help

$ fab --list   # Useful to see available tasks

$ fab hostname # Runs the hostname() function

$ fab hello   # Simple Print (but it can take a kwarg)

$ fab hello:name="Your Name"

$ fab hello:"Your Name"

Demo Code - Multiple Commands

$ fab clone_demo

    mkdir to clone code into

    run git clone

    do an `ls` on the cloned code

$ fab check_demo

    word count the fabfile.py from github

    open the file with vi

    run `git status`

Demo Code - Wait for User Input

$ fab clean_demo

    Prompt user before rm -rf

    do it if user confirms it