Continuous Integration with Jenkins
... focus on Python
Mika & Jenkins
Grml.org Project
Daily builds & Releases
(>80 Jobs)
Sipwise.com
Full release management
(>190 Jobs)
facts & FAQ
Jenkins

“The Wordpress of CI”™
Facts

Hudson: 2004
Jenkins: 2011

~weekly releases & LTS version
MIT license

>450 Plugins
>35k installations
FAQ #1: Blue?!

http://jenkins-ci.org/content/why-does-jenkins-have-blue-balls
Demo
% curl-o jenkins.war \ http://mirrors.jenkins-ci.org/war/latest/jenkins.war

% java -jar jenkins.war

% $BROWSER http://0.0.0.0:8080

NOTE: use packages for production
best practices
homogenous jobs

similar jobs share build scripts/steps
discard old builds

win performance, disk usage,..
trigger builds

... instead of polling for changes
slaves & distributed builds
scale out early
automate

... whatever hurts
external dependencies

... get rid of them
artifacts

... and fingerprints
notifications

via mail, jabber, irc,...
mobile apps
... e.g. www.jenkins-ci.mobi
backups

... and restore
useful plugins
Python Plugin
for executing python

![Execute Python script]

```python
print "huhu"
```

[workspace] $ python /tmp/hudson8488325948833298075.py
huhu
Finished: SUCCESS
## SLOCCCount Results

<table>
<thead>
<tr>
<th>File</th>
<th>Language</th>
<th>Lines</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>/debian/rules</td>
<td>makefile</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>/doc/Makefile</td>
<td>makefile</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>/mbr/mbr.S</td>
<td>asm</td>
<td>230</td>
<td></td>
</tr>
<tr>
<td>/mbr/Makefile</td>
<td>makefile</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>/grml2iso</td>
<td>sh</td>
<td>190</td>
<td></td>
</tr>
<tr>
<td>/tarball.sh</td>
<td>sh</td>
<td>84</td>
<td></td>
</tr>
<tr>
<td>/Makefile</td>
<td>makefile</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>/grml2usb</td>
<td>python</td>
<td>1022</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>1639</td>
<td></td>
</tr>
</tbody>
</table>
Violations Plugin for pep8, pylint,...
Nose Tests, Coverage,..

... see upcoming Lightning Talk :)
Timestamper Plugin

18:09:54 Started by user Michael Prokop
18:09:54 Building in workspace /var/lib/jenkins/jobs/grml2usb-source-tests/
18:09:54 Checkout:workspace /var/lib/jenkins/jobs/grml2usb-source-tests/workspace
18:09:54 Using strategy: Default
18:09:54 Last Built Revision: Revision 22238432167d60b8c1e781be26095995eb5d
18:09:54 Checkout:source /var/lib/jenkins/jobs/grml2usb-source-tests/workspace
18:09:54 Fetching changes from 1 remote Git repository
18:09:54 Fetching upstream changes from git://github.com/grml/grml2usb.git
18:09:54 Seen branch in repository origin/HEAD
18:09:54 Seen branch in repository origin/master
18:09:54 Seen branch in repository origin/mika/grml2iso-templates
18:09:54 Seen branch in repository origin/mika/media_path
18:09:54 Seen branch in repository origin/mru/grml2iso-templates
18:09:55 Commencing build of Revision 22238432167d60b8c1e781be26095995eb5de
18:09:55 Checking out Revision 22238432167d60b8c1e781be26095995eb5de52a (or
18:09:55 Warning: There are multiple branch changesets here
18:09:55 [workspace] $ /bin/sh -xe /tmp/hudson8119103293049157429.sh
18:09:55 + rm -rf reports
18:09:55 [workspace] $ /bin/sh -xe /tmp/hudson3986771262421641804.sh
18:09:55 + mkdir -p reports
18:09:55 + /usr/bin/sloccount --duplicates --wide --details source
18:09:55 [workspace] $ /bin/sh -xe /tmp/hudson9165592941863928227.sh
18:09:55 + mkdir -p reports
18:09:55 + pep8 --repeat --ignore E501 source/grml2usb
18:09:55 [workspace] $ python /tmp/hudson6260668113947609498.py
18:09:55 huhu
18:09:55 Finished: SUCCESS
Test Anything Protocol

for python, ruby, bash,...
Bruce Schneier [Plugin]... knows Alice and Bob's shared secret.
Chuck Norris [Plugin]

... can instantiate an abstract class
persona plugin
python related projects
Python API
https://launchpad.net/python-jenkins
Welcome to Python Jenkins’s documentation!

Python Jenkins is a library for the remote API of the Jenkins continuous integration server. It is useful for creating and managing jobs as well as build nodes.

Example usage:

```python
j = jenkins.Jenkins('http://your_url_here', 'username', 'password')
j.get_jobs()
j.create_job('empty', jenkins.EMPTY_CONFIG_XML)
j.disable_job('empty')
j.copy_job('empty', 'empty_copy')
j.enable_job('empty_copy')
j.reconfig_job('empty_copy', jenkins.RECONFIG_XML)

j.delete_job('empty')
j.delete_job('empty_copy')

# build a parameterized job
j.build_job('api-test', {'param1': 'test value 1', 'param2': 'test value 2'})
```

Python Jenkins development is hosted on Launchpad: https://launchpad.net/python-jenkins

Installing

```bash
pip:

    pip install python-jenkins
```

```bash
easy_install:
```
Autojenkins

https://github.com/txels/autojenkins
Welcome to AutoJenkins’s documentation!

AutoJenkins is a tool to automate or remote-control Jenkins. You can e.g. create and delete build jobs, trigger builds, read latest build results, etc.

Contents:

- Introduction to AutoJenkins
  - AutoJenkins as API
  - AutoJenkins from the Command Line
  - More Info
- autojenkins.jobs

Sources can be found at https://github.com/txels/autojenkins

Indices and tables

- Index
- Module Index
- Search Page
ShiningPanda 101

This tutorial is meant to teach you how to setup continuous integration for a Python project using ShiningPanda. ShiningPanda is based on Jenkins, but this tutorial assume no previous experience with it.

For this tutorial, we are going to build pygments, the python syntax highlighter tool. The only prerequisite is to have a ShiningPanda account.

Create a new Job

At first let’s create a new job dedicated to pygments. In your ShiningPanda hosted Jenkins instance, click on New Job.

- Build a Python project
  - This is the central feature of ShiningPanda. ShiningPanda will build your Python project, combining any SCM with some Python build environment.

- Build a maven2/3 project
  - Build a maven2 project. Jenkins takes advantage of some plugins which reduces the configuration.

- Build multi-configuration project
  - Suitable for projects that need a large number of builds or testing on multiple environments, platform-specific.

- Copy existing Job
  - Copy from

  OK
pbundler

Bundler for Python

https://github.com/zeha/pbundler
version 1.455

UI improvements
Plugins in Ruby via JRuby

Charles Lowell++
Any volunteers for Jython? :)
Ressources
Jez Humble & David Farley: Continuous Delivery, Addison-Wesley, 2010
Jenkins
The Definitive Guide

John Ferguson Smart
Foreword by Kohsuke Kawaguchi, creator of Jenkins/Hudson

wakaleo.com/books/jenkins-the-definitive-guide
Questions || Wishes?

@mikagrml
mika @ github
michael-prokop.at/blog/