Continuous Integration and Jenkins

Petr Jirout

jirout.petr1@gmail.com
Outline

• What is Continuous Integration?

• What is Jenkins?

• What can we do with both in CERN?
Outline

• What is Continuous Integration?

• What is Jenkins?

• What can we do with both in CERN?

• ... and Czech accent.
Continuous Integration

- Find a bug (broken code) as soon as possible.
- Automating as much as possible.
- Establish proper test suite.
- Monitoring and reporting.
Continuous Integration

- Set up Continuous Integration service.

- Commit changes frequently.

- Verify every change.

- Take appropriate action.
Continuous Integration

- Set up Continuous Integration service.

- Commit changes frequently.

- Verify every change.

- Take appropriate action.
Continuous Integration

- Set up Continuous Integration service.
- Commit changes frequently.
- Verify every change.
- Take appropriate action.
Continuous Integration and Jenkins
Jenkins

• Open source tool written in Java.
• Extensible via plugins.
• Easier to maintain regular builds with history.
• Easier to run basically same build with different configuration.
Jenkins

• Open source tool written in Java.
• Extensible via plugins.
• Easier to maintain regular builds with history.
• Easier to run basically same build with different configuration.

• But in the end it is just a smart scheduler.
Continuous Integration and Jenkins

Integration Tests (run when you start a QA push)

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>network-tests</td>
<td>23 min &gt; 15 hr</td>
<td>11 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>platform</td>
<td>30 min &gt; 5.5 hr</td>
<td>7 min 31 sec</td>
</tr>
</tbody>
</table>

Unit Tests (run on every commit)

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>unit-tests</td>
<td>1.6 hr &gt; 11 hr</td>
<td>15 sec</td>
</tr>
</tbody>
</table>

Functional Tests (run when a QA push is complete)

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>checkout-functional</td>
<td>29 min &gt; 1.1 days</td>
<td>3 min 5 sec</td>
</tr>
</tbody>
</table>

Smoke Tests (run when a push is complete)

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>restapi-princess-smoker</td>
<td>16 min &gt; 14 hr</td>
<td>2 min 19 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>restapi-prod-smoker</td>
<td>1 hr &gt; 4.9 hr</td>
<td>2 min 43 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>restapi-api-smoker</td>
<td>29 min &gt; 15 hr</td>
<td>34 sec</td>
</tr>
</tbody>
</table>

Etsy's Continuous Integration Engine

Bah weep gragnah weep mini bong!

Trunkage

1. Always Be Pushing
2. Tryserver status
3. Etsy PHP Documentation
4. PHP Code Duplication
5. Unit Test Coverage
   - Platform
   - Image Storage
6. Desproutening

Deployment

1. Deploystorator
   1. deploys per day
   2. time from commit to deploy
2. Deployment Dashboard - realtime production Web stats
3. SuperGrep - production logs in realtime

Artisanal Software Tools

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>platform-code-style</td>
<td>1.5 hr</td>
<td>1 min 27 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>platform-code-duplication</td>
<td>20 hr &gt; 2.8 days</td>
<td>1 min 15 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>unit-tests-coverage</td>
<td>1.9 hr</td>
<td>1 min 37 sec</td>
</tr>
</tbody>
</table>

Naughty Tests (run periodically)

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>sleep-tests</td>
<td>3.7 hr</td>
<td>1 min 37 sec</td>
</tr>
</tbody>
</table>

Tests for Etsy Services

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Job</th>
<th>Last Statuses</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>analytics</td>
<td>41 hr &gt; 31 days</td>
<td>50 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>search</td>
<td>2.8 hr &gt; 2.9 days</td>
<td>1 min 26 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sprouter</td>
<td>22 hr &gt; 1.9 days</td>
<td>1 min 55 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vashi</td>
<td>23 hr &gt; 1.5 days</td>
<td>2 min 51 sec</td>
</tr>
</tbody>
</table>
Continuous Integration in CERN

- Where?
Continuous Integration in CERN

- Where?

- In every software project.
Continuous Integration in CERN

• Where?

• In every software project.

• Like ROOT, Geant4, CernVM, CernVM-FS ...
Example – LCG externals

- Different compilers: gcc, icc

- Different architectures: x86, x86_64

- Different branches: Experimental, Release, Preview
Example – LCG externals

- Different compilers: gcc, icc
- Different architectures: x86, x86_64
- Different branches: Experimental, Release, Preview

- How to build all these possibilities?
Dealing with many configurations

- By computing power and smart configuration!
- Having pools of build machines.
  - Pets vs Cattle
- Braining up your configuration scripts.
Bring these ideas to Jenkins

• Create pools.

• Separate the environment from building process.
  – Put environment settings to one place (script).
Bring these ideas to Jenkins

• Identify the 'real variables'.
  – You have to specify them manually.

• Put these variables to configuration axes.
  – Exclude unwanted combinations.

• Let Jenkins do its job!

• Gather results.
Summary

- Continuous Integration
  - Generally good principle, but comes with a price.
  - Not only building, but also testing.

- Jenkins
  - Helps with menial tasks.
  - Does not solve your CI within single click.
What to do next

- Study!

- **Continuous Integration:**
  
  Martin Fowler's article
  
  [http://martinfowler.com/articles/continuousIntegration.html](http://martinfowler.com/articles/continuousIntegration.html)

  Paul Duvall: Continuous Integration
  

- **Continuous Delivery:**

  Jez Humble, David Farley: Continuous Delivery
  