

Teaching an Old Python New Tricks

Eggs, PyPI and Buildout

Davis Plone Users Group, 2008-03-26

Steve McMahon

Reid-McMahon, LLC

The Problem:

- Amazing things are happening in Python World
- Zope/Plone communities were isolated from them:
 - Couldn't benefit
 - Couldn't contribute

The Solution

- Zope 3 Component Architecture
 - No magic Product space
 - Components as Real Python Modules
 - ZCML is glue

Review of Python Package Installation

- Download a tarball; unpack it
- Execute `setup.py build/install` with target python
- Modules copied to `site-packages`

So, what's all this about eggs?

- Eggs use `easy_install`
- Eggs discoverable/downloadable TTW
PYPI
- Eggs may be binary
- Eggs may be **activated/deactivated**

Why does that matter (activation/deactivation)?

- Python modules can and do conflict.
- Anyone installed ArchGenXML lately?
- Demo: site-packages with eggs.

Making all that really useful: zc.buildout

- From README:

The Buildout project provides support for creating applications, especially Python applications. It provides tools for assembling applications from multiple parts, Python or otherwise. An application may actually contain multiple programs, processes, and configuration settings.

Creating a Buildout

- The tools:

- Paster

- ZopeSkel

- ```
$ paster create -t plone3_buildout myproject
$ cd myproject; python bootstrap.py
$ vi buildout.cfg
$ bin/buildout
```

# Anatomy of buildout.cfg

- Parts, Recipes, Options

- `[buildout]`  
`parts=plone`  
`zope`

```
[plone]
recipe = plone.recipe.plone <= 3.1.999
[zope2]
recipe = plone.recipe.zope2install
```

# Common Parts

```
◎ parts =
 plone
 zope2
 productdistros
 instance
 zopepy
```

# Common [buildout] Options

```
newest = false
parts =
 plone
 zope2
 ...

Add additional egg download sources here.
dist.plone.org contains archives
of Plone packages.
find-links =
 http://dist.plone.org
 ...

Add additional eggs here
elementtree is required by Plone
eggs =
 elementtree
 FeedParser

Reference any eggs you are developing here, one per line
e.g.: develop = src/my.package
develop =
```

# In the Buildout

```
o -rw----- .installed.cfg
 -rw-r--r-- README.txt
 drwxr-xr-x bin
 -rw-r--r-- buildout.cfg
 drwxr-xr-x develop-eggs
 drwxr-xr-x downloads
 drwxr-xr-x eggs
 drwxr-xr-x parts
 drwxr-xr-x products
 drwxr-xr-x src
 drwxr-xr-x var
```

# ./parts

|            |                |
|------------|----------------|
| drwxr-xr-x | instance       |
| drwxr-xr-x | plone          |
| drwxr-xr-x | productdistros |
| drwxr-xr-x | zope2          |

# More Parts, 1

- `[productdistros]`  
`recipe = plone.recipe.distros`  
`urls =`  
`nested-packages =`  
`version-suffix-packages =`

# More Parts, 2

```
• [instance]
 recipe = plone.recipe.zope2instance
 zope2-location = ${zope2:location}
 user = admin:admin
 http-address = 8080
 #debug-mode = on
 #verbose-security = on

 # If you want to register ZCML slugs
 # for any packages, list them here.
 # e.g. zcml = my.package my.other.package
 zcml =
```