

zc.buildout

OSCON'08

Tarek Ziadé <tarek.ziade@ingeniweb.com>

Who am I & what we do

CTO at Ingeniweb (Paris)

Ingeniweb - part of Alterway

Alterway - 100% open source company

Python, Zope and Plone

My work

QA

Building tools and conventions for developers

Releasing & deployment processes

Books I wrote

“Programmation Python”

Editor: Eyrolles (Paris) - 2006 - French

“Petit guide à l’usage du développeur agile”

Editeur Dunod (Paris) - 2007 - French

“Expert Python programming”

Packt Publishing (UK) - Available in September 2008

Open Source stuff

Involved in Zope and Plone

A few contributions in Python

45 min is too short



Plone drinkup tonight

Lucky Lab in NW, 1945 NW Quimby
at 6 pm

5 hours in 2006

5 hours in 2006

install python extra packages

get zope

install zope

create an instance

get extra products

read extra products doc

install extra products dependency

install extra products

doesn't work

ahhh right, install python-ldap

checkout products in development

doesn't work

ahhh right, wrong python-ldap version

start to work



5 minutes in 2008

5 minutes in 2008

get the “buildout”

```
$ python bootstrap.py
```

```
$ bin/buildout
```

start to work



Yes !!!



Yepeee !!!



hahaha !!!

zc.buildout

is a

a Coarse-Grained Python-Based Configuration-Driven Build Tool*

* definition stolen on Grok website

Paver

POSAV (Plain Old Setuptools And Virtualenv)

IDNTS (I Don't Need That Stuff)

Maven Capistrano, Ant, etc.,

Community

Zope 3

Plone

Grok

Herder (Creative Commons)

...

Our customers

All Zope based apps

Pylons based apps

Twisted based apps

Buildbot thingies

Every Python Project

zc.buildout

=

distutils + setuptools + PyPI + configuration file

distutils

=

builds and distributes a package
registers and uploads it to PyPI

get the screencast at

<http://ziade.org/oscon/demo1.m4v>

setuptools

=

simple dependencies management, namespaced package

egg distribution

provides **easy_install**

setuptools + distutils
=
the future distutils

egg

=

binary or source distribution
extended metadata
zip archive

`easy_install my_package`

=

will *get* and install “my_package”
from “PyPI”

PythonPackageIndex

=

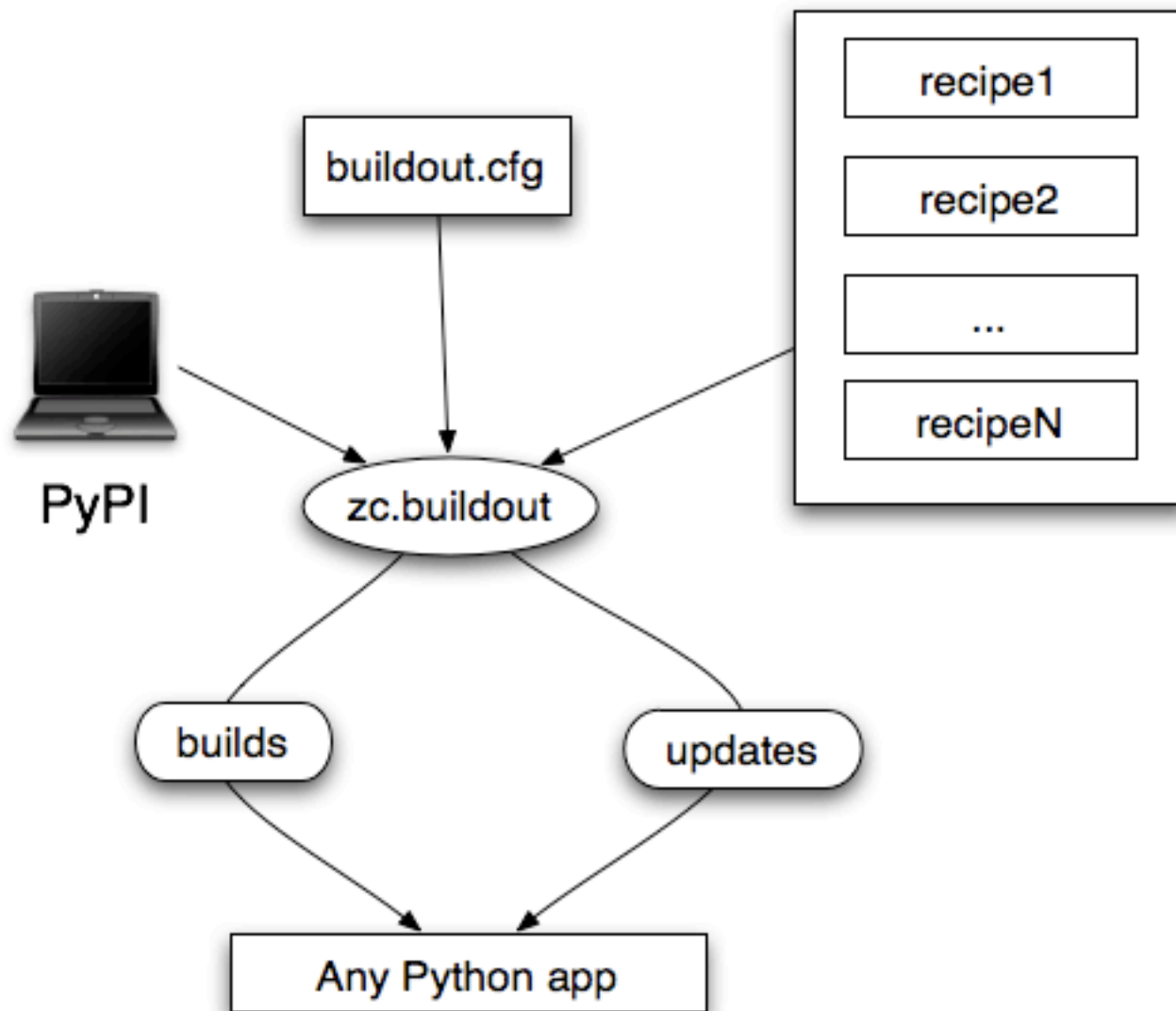
Python's CPAN, driven with distutils

register and upload commands

get the screencast at

<http://ziade.org/oscon/demo2.m4v>

A zc.buildout primer



buildout.cfg : [simple ini file](#) (ConfigParser)

```
[buildout]
```

1

```
parts =  
    one  
    two
```

2

```
[one]  
recipe = my.recipe.one  
arg1 = value1
```

3

```
[two]  
recipe = my.recipe.two  
arg1 = value1  
arg2 = value2
```

buildout.cfg processed

- 1 process each part sequentially
- 2 looks for “my.recipe.one”
gets it
executes it
- 3 look for “my.recipe.two”
gets it
executes it

demo

installing `zc.buildout`
creating a simple buildout
playing with it

get the screencast at

<http://ziade.org/oscon/demo3.mov>

demo debrief

buildout.cfg describes the application
recipes = simple packages
change the .cfg -> re-run bin/buildout
zc.buildout grabs dependencies

What we get

bin

- buildout

- scripts from recipes

eggs

- all dependencies

develop-eggs

parts

- data from recipes

A recipe primer

a class + an entry point = a recipe

a recipe = a simple package

buildout developers love **Paste Script**
Plone community provides ZopeSkel

demo

writing a recipe using a Paste template
setting an entry point
publishing it at PyPI

get the screencast at

<http://ziade.org/oscon/demo4.mov>

Hundreds of recipes at PyPI

Trove classifier : **Framework :: Buildout**

<http://pypi.python.org/pypi?:action=browse&show=all&c=512>



great !!!

Phone application lifecycle

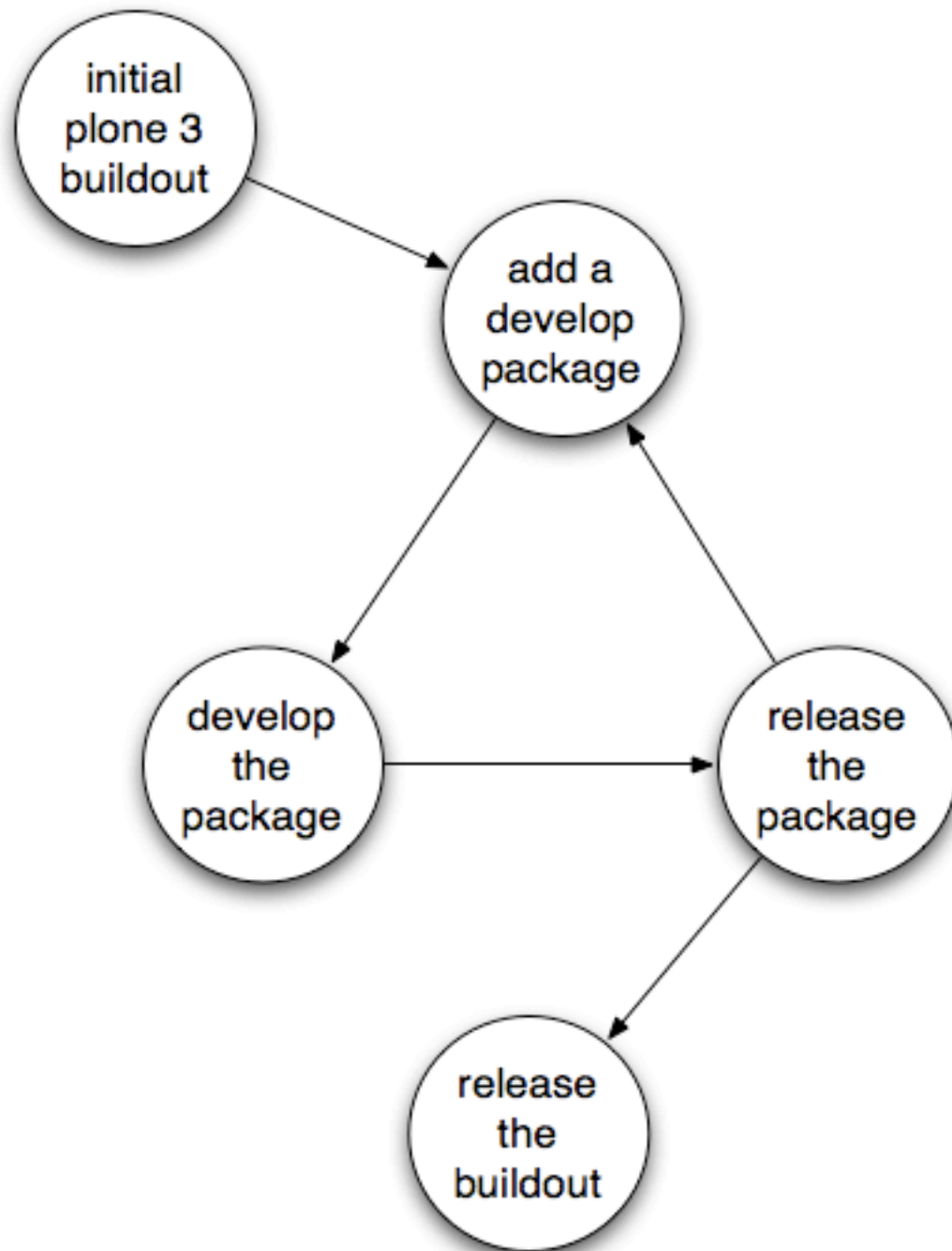
Warning

Plone buildouts are source based

Windows developers

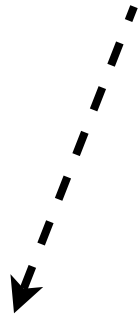
Get my Windows installer : `python2.4.4-win32.zip`

Google “An installer for a buildout-ready Windows”

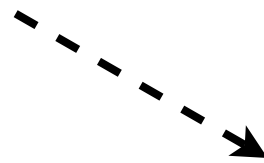


[buildout]

```
develop =  
    path/to/my/package1  
    path/to/another/package
```



```
cd path/to/my/package1  
python setup.py develop
```



buildout/develop-eggs

Typical buildout layout uses the **extends** feature

buildout.cfg

dev.cfg (extends buildout.cfg)

prod.cfg (extends buildout.cfg)

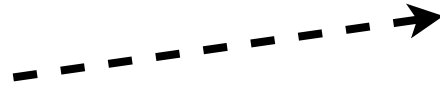
+ bootstrap.py

buildout.cfg :

```
[buildout]  
parts =  
    one  
    two
```

dev.cfg :

```
[buildout]  
extends = buildout.cfg  
parts =  
    three
```



```
[buildout]  
parts =  
    one  
    two  
    three
```

demo

creating a fresh Plone 3 buildout (Paste)

adding the dev.cfg

hooking a new development package

adding a prod.cfg

get the screencast at

<http://ziade.org/oscon/demo5.mov>

complex example: plone.org

<https://svn.plone.org/svn/plone/PloneOrg/buildouts/trunk/devel/>



varnish

zope

zeo

python-ldap

pound

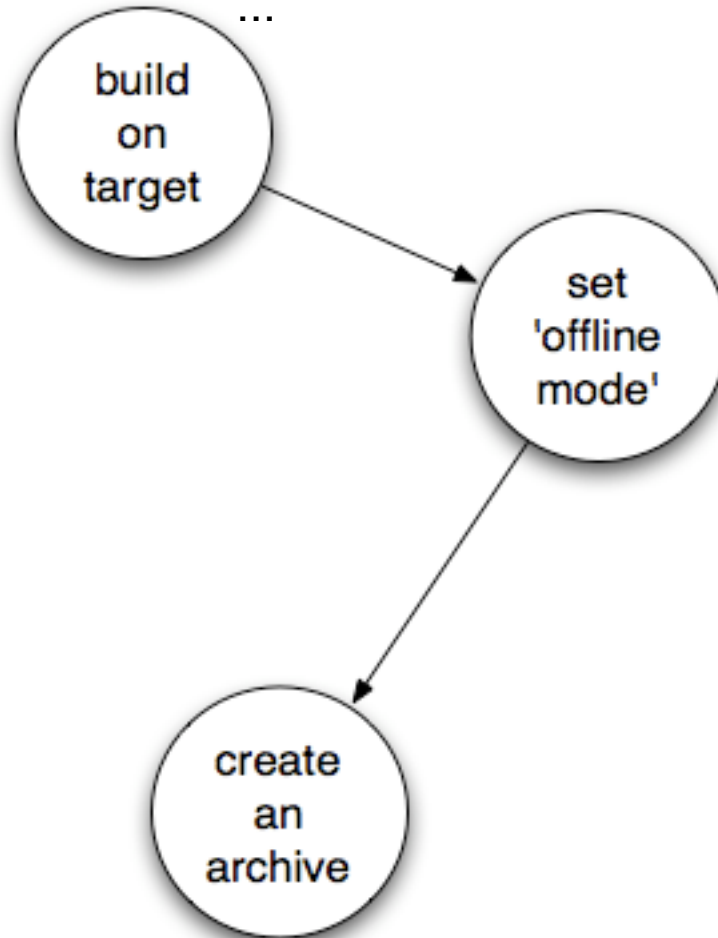
<http://tinyurl.com/plone-buildout>

Releasing the application

Source distribution

[buildout]
download-cache = downloads

...



[buildout]
offline = true
...

zc.buildout caveats

Hard to do TDD for recipe code

isolated system = boundary with os is fuzzy

Tools on the top of `zc.buildout`

`collective.releaser`

scripts to automate buildout releases and update

`collective.buildbot`

buildout recipe to create a buildbot

get the screencast at

<http://ziade.org/oscon/demo6.mov>



amazing !!!

alternative eggs locations

Why ?

PyPI == SPOF

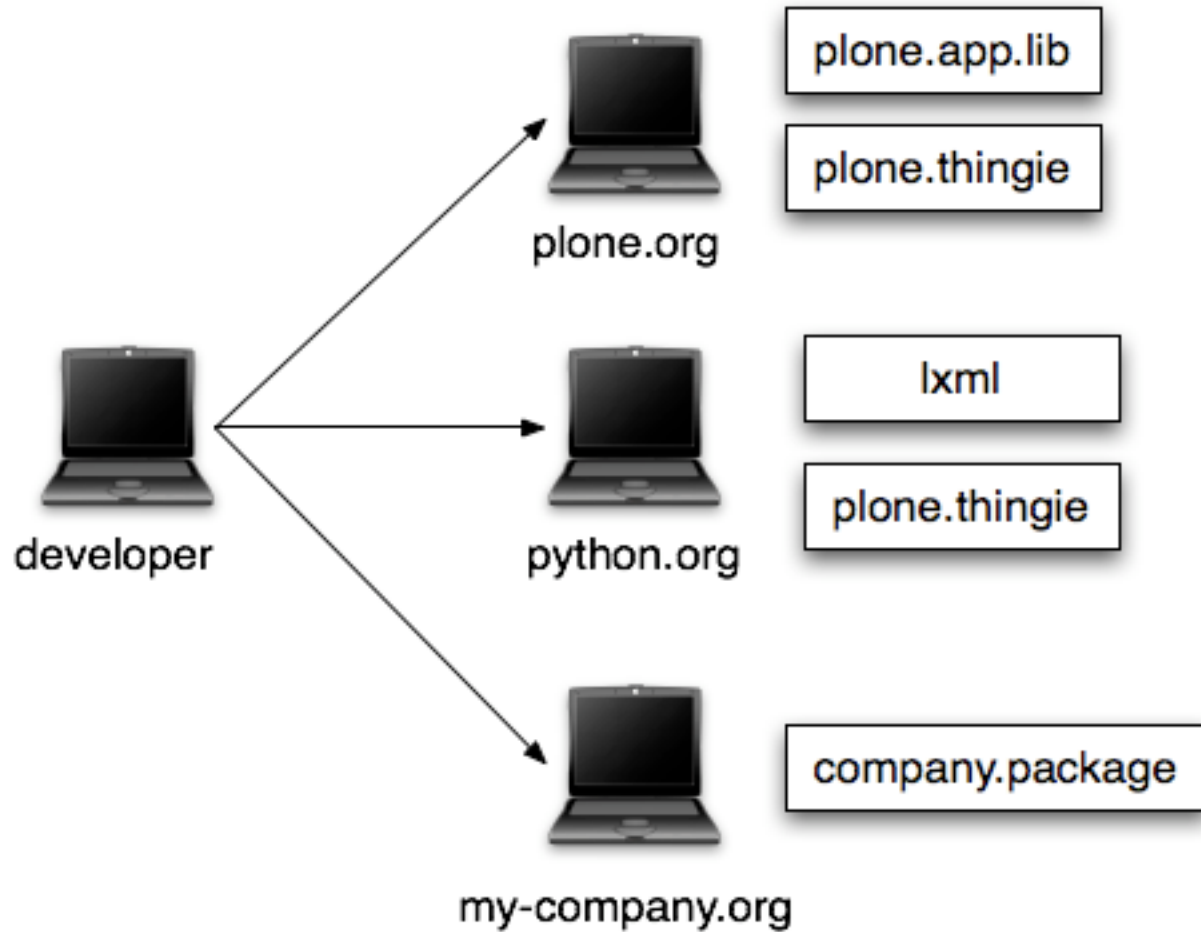
Eggs are not on PyPI sometimes

Eggs need privacy sometimes

A “Company PyPI” is useful

Command line releasing is cool

Distributed model



How : the find-links option

[buildout]

find-links =

<http://plone.org/products>

<http://my-company/eggs>



a simple directory

a directory published with a web server

or...

PloneSoftwareCenter

PloneSoftwareCenter

same APIs than PyPI
distutils compliant
powers the next plone.org
can be used for any package

That's all folks !

Thanks, questions ?

tarek.ziade@ingeniweb.com

<http://tarekziade.wordpress.com>

zc.buildout @ PyPI

collective.buildbot @ PyPI

collective.releaser @ PyPI

<http://alterway.fr>

Diagrams made with OmniGraffle (it rocks)

The Alter Way approach

Mission : bring a long-lasting and mature business model to Open Source

Significant contributions to Free Software

<http://www.alterway.fr/>

Federation of market leaders to foster the most complete solution to all IT needs for SME and institutions

The first european integrator 100% Open Source to be a real alternative to traditionnal players

Revenue in 2008 : 8 M€ (target in 2011 : 25 M€)
Source in France