



Brasília, 25 de abril de 2012

# Buildout na prática

Clayton Caetano de Sousa

# Buildout

- É simples de usar e de manter.
- Evita conflitos com os pacotes instalados.
- Instala as dependências dos pacotes.
- Não instala pacotes no Python do sistema.
- Consegue instalar: Banco de Dados, Web Cache etc.

# Vamos começar !

O que vamos fazer:

- Criar um buildout para o Plone 4.1.4.
- Instalar produtos.

# Instalando pacotes no Ubuntu

→ O Plone 4.1.4 necessita do python2.6, então vamos instalar:

```
$ sudo apt-get install python2.6 python2.6-dev python-virtualenv
```

# Instalação pacote/bibliotecas no Ubuntu

→ Indexação e compiladores:

```
$ sudo apt-get install lynx tidy xsltproc xpdf wv build-essential
```

→ Geração de thumbnails (PIL ou Pillow):

```
$ sudo apt-get install libjpeg62-dev libpng12-dev  
libfreetype6-dev libgif-dev
```

→ Pacotes para ldap, lxml

```
$ sudo apt-get install libssl-dev libsasl2-dev libldap2-dev  
libxml2-dev libxslt1-dev
```

# Python virtual

- Criando um Python separado para evitar instalar pacotes no Python do sistema:

```
$ mkdir worldploneday
```

```
$ cd worldploneday
```

```
$ virtualenv --python=python2.6 -v --no-site-packages Python2.6
```

# Instalando o ZopeSkel

```
$ ./Python2.6/bin/easy_install ZopeSkel
```

```
$ ./Python2.6/bin/paste create --list-templates
```

```
plone4_buildout:  A buildout for Plone 4 developer  
installation
```

```
$ ./Python2.6/bin/paster create -t plone4_buildout plone-  
buildout
```

```
$ cd plone-buildout
```

```
$ ../Python2.6/bin/python bootstrap.py
```

# Opções de execução

→ “-v” e “-q”

- ♦ Aumenta e diminui o nível de detalhamento.

→ “-n” e “-N”

- ♦ Procura novas versões e não procura.

→ “-O” e “-o”

- ♦ Modo on-line e off-line.

→ “-t”

- ♦ Define um timeout.



# Visualizando e Executando

→ Abrindo o “buildout.cfg”

```
$ gedit buildout.cfg
```

→ Executando

```
$ ./bin/buildout -v -t 30
```

# Configuração

## → Caracteres reservados

```
: $ % ( )
```

## → Variáveis

```
${<part_name>:<option_name>}
```

```
${buildout:parts-directory}
```

```
eggs = ${instance:eggs}
```

# Praticando

- Instalando produtos.
- Alterando a porta “http”.
- Habilitando “debug-mode” e “verbose-security”.
- Adicionando “effective-user”.
- Adicionando “zope-conf-additional”.
- Adicionando “zodb-temporary-storage”.
- Adicionando “environment-vars”.
- Definindo versão dos módulos e produtos.

# Utilizando comandos do SO

- É possível executar comandos nos SO com “**plone.recipe.command**”. Ex:

```
[buildout]
parts =
commands
...

[commands]
recipe = plone.recipe.command
command =
    export ENV_PATH=${buildout:directory}/../
    echo "import sys; sys.setdefaultencoding('utf-8')" >
    $ENV_PATH/lib/python2.6/sitecustomize.py
```

# Links

- <http://buildout.org>
- <http://planet.plone.org>
- <http://pypi.python.org/>
- <http://plone.org/documentation/tutorial/buildout>
- <http://www.sixfeetup.com/swag/buildout-quick-reference-card>
- <http://www.simplesconsultoria.com.br/blog/buildout-para-oque-der-e-vier>
- <http://collective-docs.readthedocs.org/en/latest/troubleshooting/buildout.html?highlight=buildout>

# Dificuldades/Dúvidas

- Lista de discussão sobre zope/plone:  
[zope-pt@yahoo grupos.com.br](mailto:zope-pt@yahoo grupos.com.br)