python

```
import os, shutil, sys, tempfile, urllib2

tmpeggs = tempfile.mkdtemp()

try:
    import pkg_resources
except ImportError:
    ez = {}
    exec urllib2.urlopen('http://peak.telecommunity.com/dist/ez_setup.py'
                         ).read() in ez
    ez['use_setuptools'](to_dir=tmpeggs, download_delay=0)

    import pkg_resources
```

apache

```
WSGIPythonHome ${sandbox}
WSGIDaemonProcess tmp threads=1 processes=4 maximum-requests=10000 python-path=${sandbox
<VirtualHost *:80>
  ServerName my.machine.local
  WSGIScriptAlias /site ${sandbox}/bin/zope2.wsgi
  WSGIProcessGroup tmp
  WSGIPassAuthorization On
  SetEnv HTTP_X_VHM_HOST http://my.machine.local/site
  SetEnv PASTE_CONFIG ${sandbox}/etc/zope2.ini
</VirtualHost>
```

shell

```
#! /bin/bash

INSTALLDIR=`dirname $0`
if [ -z "$INSTALLDIR" ] ; then
   INSTALLDIR=`pwd`; export INSTALLDIR
fi

ARG1=$1

echo
echo This installer actually builds Zenoss.
echo For a simpler installation try the VMPlayer Appliance image,
echo or use RPMs for Redhat based systems.
echo
echo Building...
echo

# interactive install (prompt for usernames and passwords)
if [ -z "$ARG1" ]; then
    exec $INSTALLDIR/build.sh
fi

# non-interactive install (use defaults)
if [ "${ARG1}" = "--no-prompt" ]; then
    exec $INSTALLDIR/build-noprompt.sh < /dev/null
fi
```

traduction

```
#. Default: "Switch between visual editor and HTML view"
#: kupu/plone/kupu_plone_layer/kupu_wysiwyg_support.html:189
msgid "toggle_source_view"
msgstr "modifier le code HTML"
```

config

```
[buildout]
parts =
    rst2pdf
find-links =
#reportlab
    http://ftp.schooltool.org/schooltool/eggs/3.4
#wordaxe
    http://sourceforge.net/project/platformdownload.php?group_id=105867

[rst2pdf]
recipe = zc.recipe.egg:scripts
eggs =
    rst2pdf
    simplejson
    wordaxe
```

javascript

```
WidgeteerDrawerTool.prototype.closeDrawer = function(button) {
    if (!this.current_drawer) {
        return;
    };
    this.current_drawer.hide();
    this.current_drawer.editor.resumeEditing();
    this.current_drawer = null;
    var parentdoc = parent.document;
    var placeholder = parentdoc.getElementById('drawerplaceholder')
    placeholder.style.display = 'none';
};
```

XML

```
<schema keytype="ZConfig.tests.test_schema.uppercase">
  <sectiontype name="type-2"/>
</schema>
```

HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="en-ca">
<head>
  <meta http-equiv="Content-Type"
 content="text/html; charset=windows-1252">
  <title>TwistedSNMP</title>
  <link rel="stylesheet" type="text/css" href="style/sitestyle.css">
</head>
<body
 style="background-color: rgb(255, 255, 255); color: rgb(0, 0, 0); direction: ltr;"
 alink="#008000" link="#000080" vlink="#800080">
<h1>TwistedSNMP<br>
</h1>
<p>TwistedSNMP is a set of SNMP protocol implementations for Python's
Twisted Matrix networking framework using the PySNMP project.  It
provides the following:</p>
<ul>
  <li>get, set, getnext and getbulk Manager-side queries</li>
  <li>get, set, getnext and getbulk Agent-side services</li>
</ul>
<p>Eventual goals of the system:<br>
</body>
</html>
```