
SciELO Books Documentation

Release 1rc7

SciELO

November 18, 2014

1	Documentation Index	1
1.1	How to install a SciELOBooks instance	1
2	Support	5

Documentation Index

1.1 How to install a SciELOBooks instance

1.1.1 Install pre-requisites

Before installing the SciELO-Books application, install the software listed below.

Server software

software	product URL	installation method	Ubuntu Package name
Python 2.7	http://www.python.org/	OS package manager	python2.7
python2.7	http://www.python.org/	OS package manager	python2.7-dev
GNU compiler and tools	http://www.python.org/	OS package manager	build-essential
Apache HTTP server 2.2	http://httpd.apache.org/	OS package manager	apache2
mod_wsgi - Python adapter(optional)	http://code.google.com/p/modwsgi/	OS package manager	libapache2-mod-wsgi
CouchDB 1.0	http://couchdb.apache.org/	OS package manager	couchdb
PostgreSQL	http://www.postgresql.org/	OS package manager	postgresql
GIT	http://git-scm.com/	OS package manager	git-core
Subversion	http://subversion.apache.org/	OS package manager	subversion

1. Install each package below using the recommended installation method above.

Note: Python comes pre-installed in most Linux distributions. If Python 2.5 or 2.6 is already installed, there is no need to install a newer version.

Note2: on Ubuntu 10.04 (Lucid) build-essential includes: dpkg-dev, g++, libc6-dev and make

System-wide Python libraries

software	product URL	installation method
distribute 0.6.10	http://pypi.python.org/pypi/distribute	<code>sudo python distribute_setup.py</code>
virtualenv	http://pypi.python.org/pypi/virtualenv	<code>sudo easy_install virtualenv</code>
python gxf module	http://www.swftools.org/gfx_tutorial	Installation instruction topic 1.1 Compiling gxf and installing

2. Download the `distribute_setup.py` script and use the installed Python interpreter to run it as root (this provides the `easy_install` utility):

```
# wget http://python-distribute.org/distribute_setup.py
# python distribute_setup.py
```

3. Use `easy_install` to download and install `virtualenv`:

```
# easy_install virtualenv
```

1.1.2 Configure the database

4. Create a `scielobooks_1a` database. The user name and password will not be configured at this time.

Log in to the Futon's web interface:

`http://localhost:5984/_utils/`

1.1.3 Install the application environment

Note: all of the remaining steps can be performed by a regular user without root access.

5. Use `virtualenv` to create an application environment and activate it:

```
$ virtualenv --distribute --no-site-packages scielobooks-env
$ source scielobooks-env/bin/activate
(scielobooks-env)$ # note that the shell prompt displays the active virtual environment
```

1.1.4 Install the scielobooks application

6. Go to a suitable installation directory and check out the application source:

```
Development (Recommended) :
Read-only:
(scielobooks-env)$ git clone git://github.com/scieloorg/scielobooks.git
Read+write:
(scielobooks-env)$ git clone git@github.com:scieloorg/scielobooks.git
```

7. With the `scielobooks-env` environment active, use `setuptools` to automatically download and install all the dependencies:

```
(scielobooks-env)$ pip install -r requirements.txt

(scielobooks-env)$ python setup.py install
```

1.1.5 Running the application

Paster

The Pyramid web framework already comes with Paster. So, in order to run it you simply need to:

1. Create a paster .ini configuration file:

```
Development:
$ cp development-TEMPLATE.ini development.ini

Production:
$ cp production-TEMPLATE.ini production.ini
```

Note: The application comes with 2 base templates, for development and for production environments.

See <http://pythonpaste.org/script/#configuration> for more information about PasteScript.

2. Run:

```
$ paster serve production.ini --daemon
```

Apache with mod_wsgi

1. Create and configure a paster .ini configuration file. Note: The application comes with 2 base templates, for development and for production environments.

Development:

```
$ cp development-TEMPLATE.ini development.ini
```

Production:

```
$ cp production-TEMPLATE.ini production.ini
```

See <http://pythonpaste.org/script/#configuration> for more information about PasteScript.

2. Create and configure a .wsgi configuration file. Note: The application comes with a directory named *apache*, containing templates for deployments using Apache with mod_wsgi

Development:

```
$ cp apache/app/devel-TEMPLATE.wsgi apache/app/devel.wsgi
```

Production:

```
$ cp apache/app/production-TEMPLATE.wsgi apache/app/production.wsgi
```

Note: The .wsgi configuration file must be configured to point to the previously created .ini file, to match the application's entry point.

3. Configure the Apache WebServer Note: The application comes with 2 virtual hosts base templates. You can simply create a symlink to the apache's available sites.

Development:

```
$ cp apache/httpd-devel-TEMPLATE.conf apache/httpd-devel.conf
```

Production:

```
$ cp apache/httpd-TEMPLATE.conf apache/httpd.conf
```

See <http://docs.pylonsproject.org/projects/pyramid/1.0/tutorials/modwsgi/index.html> for more information about deploying a Pyramid app using mod_wsgi.

1.1.6 WordPress Integration

In order to both applications, the main site (Wordpress) and the details site (Python), coexist transparently, we need to add some rules in the webserver.

Basically, the catalog package must be accessible from the Wordpress domain, i.e. *http://books.scielo.org/id/w2* must resolve to *http://admin.books.scielo.org/id/w2*. The latter should not be accessible for users.

Rules to reverse proxy some requests:

```
# wordpress app virtualhost
<Proxy *>
    Allow from all
</Proxy>

ProxyPassMatch ^/staff/(.*)$ http://homolog.backoffice.books.scielo.org/staff/$1
ProxyPassMatch ^/id/(.*)$ http://homolog.backoffice.books.scielo.org/id/$1
ProxyPassMatch ^/static/(.*)$ http://homolog.backoffice.books.scielo.org/static/$1
ProxyPassMatch ^/deform_static/(.*)$ http://homolog.backoffice.books.scielo.org/deform_static/$1
ProxyPassMatch ^/setlang/$ http://homolog.backoffice.books.scielo.org/
ProxyPassMatch ^/login/(.*)$ http://homolog.backoffice.books.scielo.org/login$1
ProxyPassMatch ^/logout/(.*)$ http://homolog.backoffice.books.scielo.org/logout$1

ProxyPass /admin http://homolog.backoffice.books.scielo.org/login

ProxyPassReverse /admin http://homolog.backoffice.books.scielo.org/login
ProxyPassReverse /staff http://homolog.backoffice.books.scielo.org/staff
```

1.1.7 Troubleshooting

The application is updated but seems like the cache is not (even after the apache+mod_wsgi have been restarted)

mod_wsgi has an aggressive cache system. to refresh it, you have to update the date of the .wsgi configuration file and restart apache:

```
$ touch apache/app/production.wsgi
```

Support

- [Project Issue Tracker](#)
- [Mailing lists](#)
- [Source code](#)