Olympos

Chronos Web Tools

Installation guide 0.9.9.2

Author: Giuseppe Platania
Owner: Giuseppe Platania

Document History

Document Location

This is a snapshot of an on-line document. Paper copies are valid only on the day they are printed. Refer to the author if you are in any doubt about the currency of this document.

The source of the document will be found in the Subversion repository on Source Forge.

https://olympos.svn.sourceforge.net/svnroot/olympos/trunk/olympos/chronos/scr/applications/requirements/Chronos_Web_Modeller_Installation.pdf

Revision History

Date of this revision: 19/11/10	Date of next revision *TBD*

Revision Number	Revision Date	Summary of Changes
(#)	(-)	(Describe change)
0.1	18/12/08	created
0.2	03/02/09	Update installation steps
0.8.1	04/02/09	Update for 0.8.1
0.9	04/30/09	Update for 0.9
0.9.5	11/19/10	Update for 0.9.5
0.9.7	09/04/09	Update for 0.9.7
0.9.7.1		Review
0.9.8.4	01/11/10	Review
0.9.9	05/03/10	Added cron job section
0.9.9.2	06/16/10	Added pdo settings
0.9.9.2	07/05/10	UMI setting added

Contents

1. Abstract

This file helps to configure and install the **Chronos Web Tools,** including **the Chronos Web Modeller** and the **Chronos web Browser**. It is addressed to system administrators only. If you are an user start with the CWM online help.

You can download the most recent CWT version from: http://sourceforge.net/projects/olympos/

2. License

The **code** of the CWT and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 which accompanies this distribution, and is available at http://www.eclipse.org/legal/epl-v10.html.

The **concept** of Chronos is made available under the Creative Common license.

You are free:

* to Share — to copy, distribute and transmit the work

* to Remix — to adapt the work

Under the following conditions:

* Attribution. You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

* For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

* Any of the above conditions can be waived if you get permission from the copyright holder.

* Nothing in this license impairs or restricts the author's moral rights.

3. Requirements

Please consider the following list of requirements before installing the Chronos Web Modeler

3.1 Server hardware

Minimal configuration (development)

CPU	1x2,5 GHz
RAM	1 GB
HD	1 x 500 GB
RAID Level	0

Suggested (productive)

CPU	4 x 2,5 GHz
RAM	8 GB
HD	2 x 1000 GB
RAID Level	5

3.2 Server Software stack

The development was tested under Windows XP SP2, Linux Ubuntu Server 64 and Linux Debian (Etch 4.0 Kernelname: 2.6.18-6-686-bigmem). Other Linux distributions (like e.g SUSE or Redhat) should also work, but have not been tested .

The server requires the following components:

- Apache (2.2.11) [Linux]
- Apache (2.2.10) [Windows]
- MySQL (5.0.00)
- PHP (5.2.8) (with JSON library active)
- Java 1.6
- phpMyAdmin (3.1.1) [Linux]
- phpMyAdmin (3.1.0) [Windows]
- unzip
- OpenOffice version 2.4 until 3.0

XAMPP for Linux or Windows, includes most of them.

Download and install XAMPP for Linux or Windows from
http://www.apachefriends.org/en/xampp.html

3.3 CWM Application Libraries

The CWM application uses the libraries depicted below:

3.4 Client

The client need to be able to run a supported browser.

3.4.1 Hardware

Minimal :

Key	Value
CPU	Celeron 900 MHz Class
RAM	752 MB
Monitor	15"

Suggested:

Key	Value
CPU	CORE2 2 GHz class
RAM	2 GB
Monitor	21"

3.4.2 Software

The client runs in following browsers with activated JavaScript:

- Firefox 3 or better (*Preferred*)
- Google Chrome 3 or better

Not (yet) supported:

- Opera
- IE 8

Installation

The installation of the CWM needs following steps:

1. Install the LAMP or WAMP stack.
2. Configure PHP.
3. Install phpMyAdmin.
4. Create the CWM database.
5. Install Java.
6. Install OpenOffice
7. Deploy the CWM package.
8. Configure the application.

This document is oriented to a productive installation under Linux.

3.5 Install the LAMP or WAMP stack

Follows the specific installation instruction of your Operating System or distribution.

3.6 Configure PHP

1. find the position of your php.ini

2. edit php.ini as following:

```
;;;;;;;;;;;;;;;;;
; Data Handling ;
;;;;;;;;;;;;;;;;;
variables_order = "GPCSE"

;;;;;;;;;;;;;;;;;;;;
; Resource Limits ;
;;;;;;;;;;;;;;;;;;;;
max_execution_time = 7200
max_input_time = 7200
memory_limit = 1024M

;;;;;;;;;;;;;;;;;;;
; Fopen wrappers ;
;;;;;;;;;;;;;;;;;;;
default_socket_timeout = 300
```

```
;;;;;;;;;;;;;;;;;;;
; Module Settings ;
;;;;;;;;;;;;;;;;;;;
[Pdo_mysql]
pdo_mysql.default_socket = /opt/lampp/var/mysql/mysql.sock

[Session]
session.gc_maxlifetime = 7200
```

3.7 Install phpMyAdmin

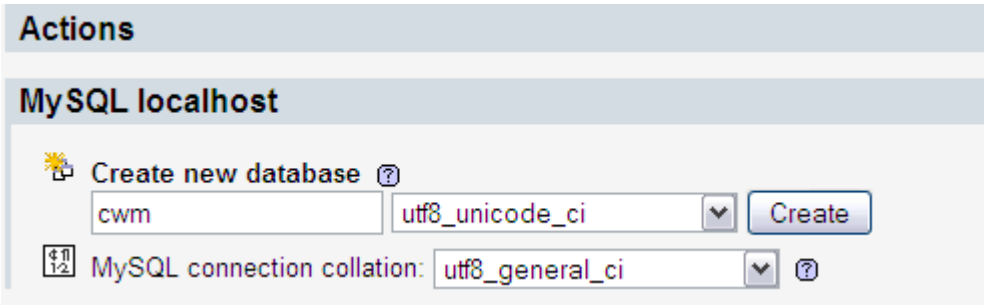
Phpmyadmin is a supported package in many Linux distribution.

Under Ubuntu you can install it by using the following command:

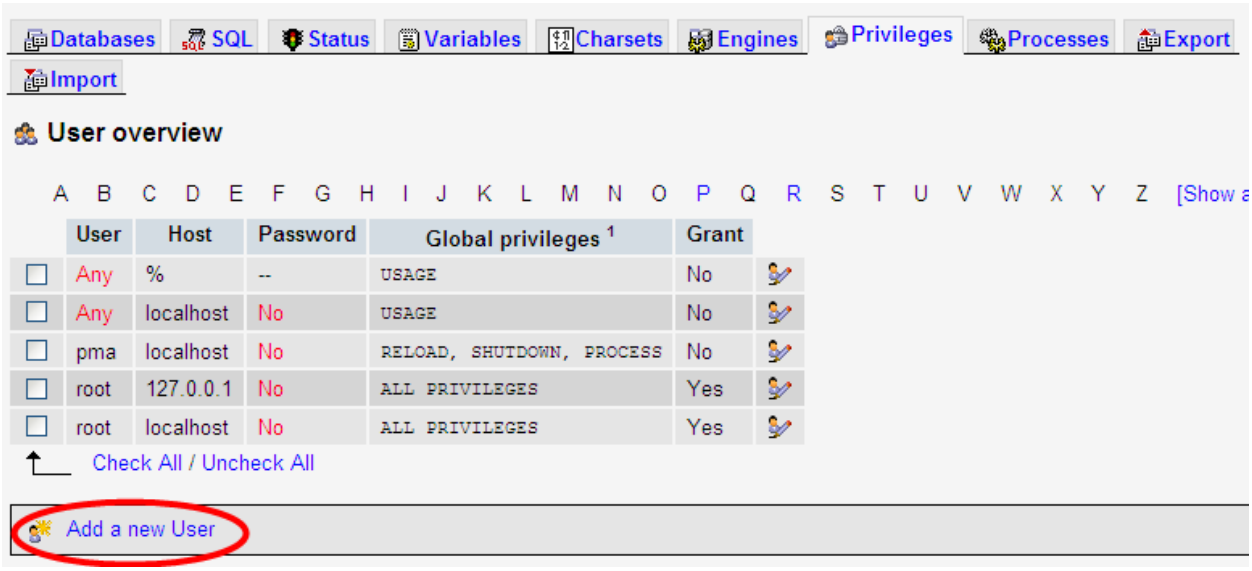
```
sudo apt-get install phpmyadmin
```

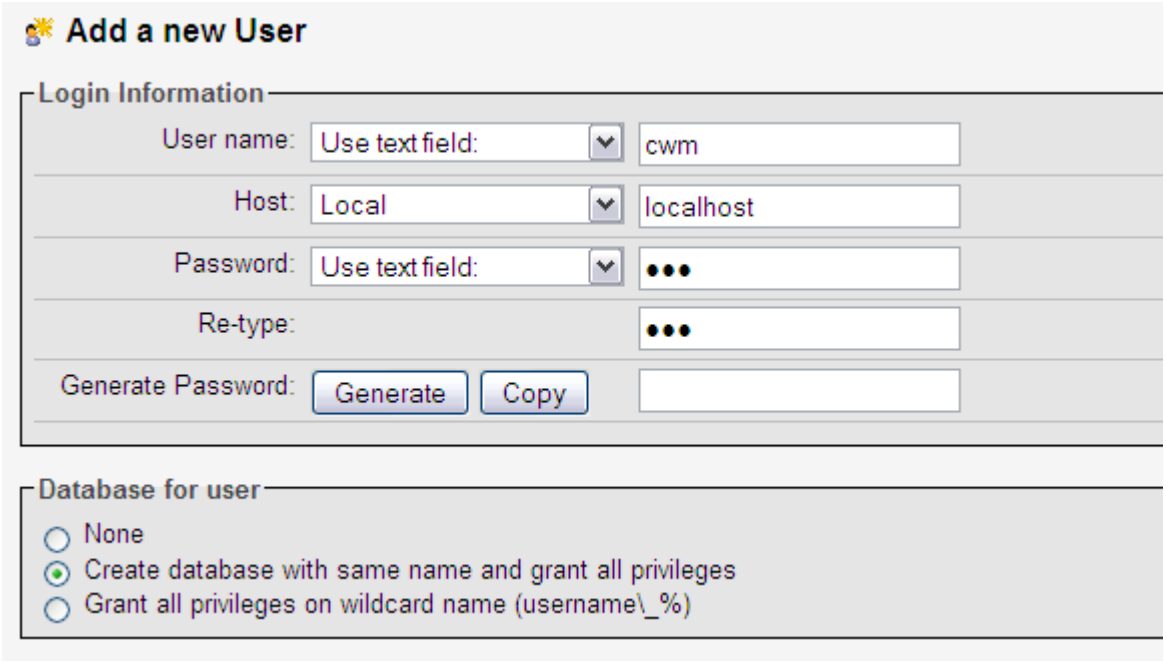
3.8 Create the CWM database

- Only in case of a new installation
- Open PHPMyAdmin by browsing to lhttp://localhost/xampp/
- Create a Database name “cwm”:
-



- Create an user with name “cwm” for the new created “cwm” database
- set the password to “cwm” all lowercase for this user





3.9 Fill the database

Only in case of a new installation:

- execute on the created CWM database the provided `cwm.sql.gz`. This will install the default users and roles as well as the standard models.

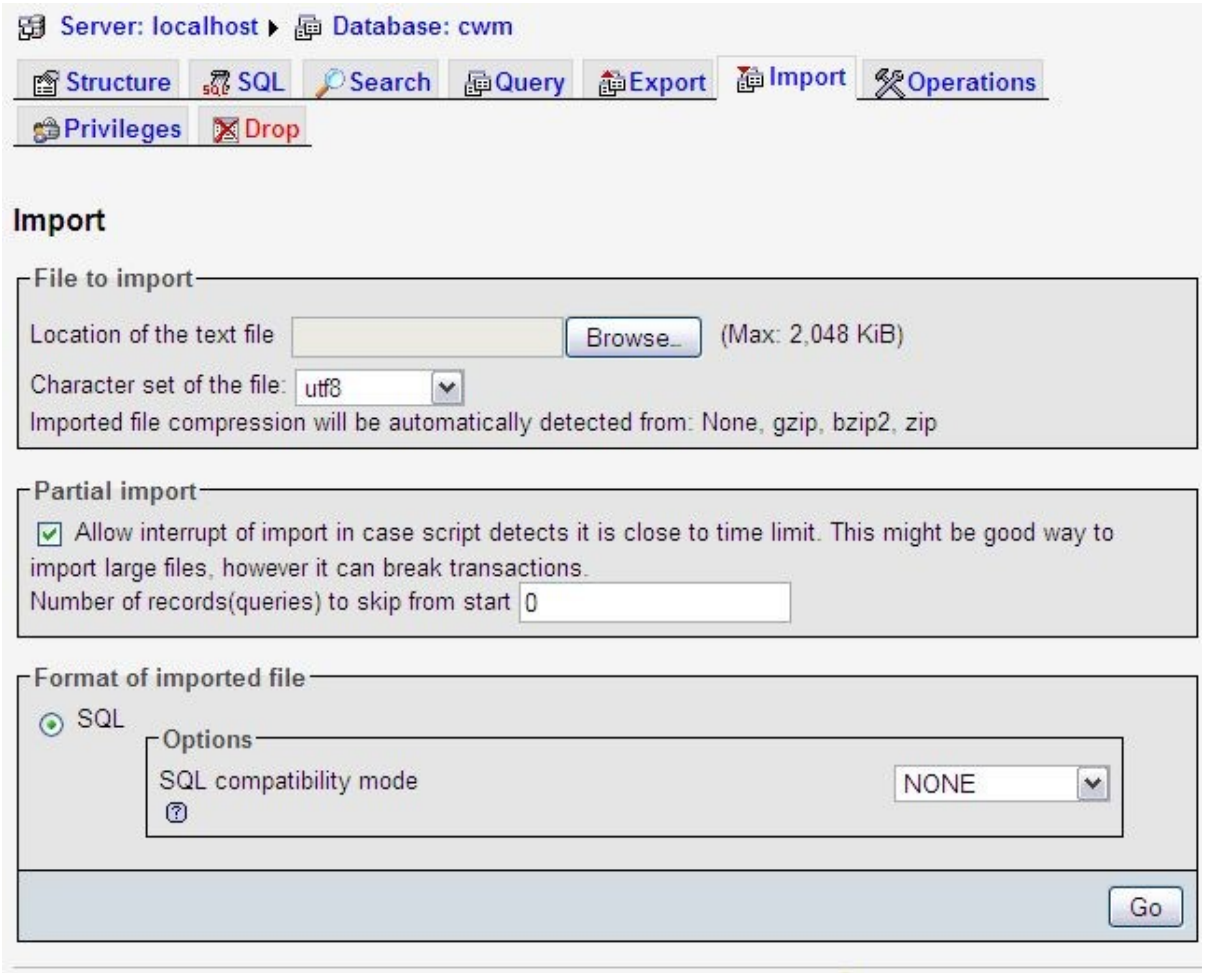


Illustration 1:

3.10 Install Java

- Check if the correct java version is installed: type in a console "`java -version`"
- If 1.6 is not available, install java 1.6 or better

3.11 Install OpenOffice

Install the package

openoffice.org

the version number must be 2.0.4 until 3.0[1].

- Type in a bash shell:
  ```
  soffice -headless -accept="socket,host=localhost,port=8100;urp;"
  -nofirststartwizard
  ```

1 The actual 0.9.9.2 release don't support

Note: if you receive the error "X11 error: Can't open display:" when running the above command, then you may need to install the "headless" OpenOffice.org package from your Linux distribution. On Ubuntu/Debian, this package is called **openoffice.org-headless**.

3.12 Test the access to the Metamodel

To be able to export the CWM need to dynamically read the Metamodel that in this version is under:

`http://cwm.nikostotz.de/chronos.profile.uml`

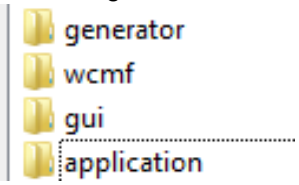
check that your server or firewall can access to this file by typing

```
wget  http://cwm.nikostotz.de/chronos.profile.uml
```

3.13 Deploy the CWM package

- The CWT are provided as single zip file.
  - **NOTE Optional in case of Upgrade only**

    remove the `server.ini` file from cwm/application/include before unzip the file to avoid overwrite your actual configuration.
- Unzip the package (e.g. cwm.zip) in the apache http directory e.g. /var/www directory
- following structure is created:



3.14 Ensure write access for web server user

Make sure the user running the web server (`www-data` on Debian / Ubuntu) has write access to the following directories:

- cwm/application/log
- cwm/application/statistics
- cwm/application/searchIndex
- cwm/application/include/views/smarty/templates_c
- cwm/application/include/views/smarty/cache

3.15 Configure the application

Only in case of a new installation:

- Browse to cwm\application\include
- Open with an editor the `server.ini` file

- Edit the section [database] with your DB information

```
[database]
dbType = mysql
dbHostName = localhost
dbName = cwm
dbUserName = cwm
dbPassword = cwm
```

3.16 Configure the standard Model language

in the same file

application/include/server.ini

configure your default language for the backend model. Use the 2 letter international code (en for English, de for German, it for Italian, ….)

```
[i18n]
defaultLanguage = de
```

now browse to

gui/js/uwm/Config.js :

to configure the front end model language.

```
...
/**
 * The default modelling.
 *
 * @type String
 */
uwm.Config.defaultLanguage = 'de';
```

3.17 Configure the generator

Only in case of a new installation:

- on the same file `server.ini` modify the `[generator]` section to point to the position of your ChronosGenerator.jar.

```
...
[generator]
executable = /var/www/cwm/generator/ChronosGenerator.jar
```

3.18 give Apache rights to the generated folder

From the version 0.9.8 the CWM can generate a complete application.

Give the Apache all the right for:

```
 /var/www/cwm/generated/
chmod 666 /var/www/cwm/generated/
```

3.19 Install cron Job

Starting with version 0.9.9, a cron job is provided for the Chronos Web Browser. It's supposed to preprocess large models on a regular basis, so the users can view the CWB contents without waiting for the full generation process.

The configuration for the cron job resides in `/var/www/cwm/application/jobs/config.ini`. The URL must point to the web path to CWM `main.php`. The login and password must be a user with read access to all models. You should create a special user for this task.

An example config.ini looks like:

```
[server]
url = http://localhost/cwm/application/main.php
login = cronuser
password = Sup3rSECretPAssWorD
```

The cron job must use `/var/www/cwm/application/jobs/` as working directory. The script to be called is `/var/www/cwm/application/jobs/statistics.php` with a PHP5 interpreter. It should be run with the web server user. The user must have write access to `/var/www/cwm/application/jobs/log`. All output of the cron job is written to this directory. For tracking installation issues, you still should monitor the standard output of the job. An example cron job file looks like:

```
HOME=/var/www/cwm/application/jobs/
25 2 * * *  www-data    /usr/bin/php5 -q
/var/www/jobs/application/jobs/statistics.php 2>&1 >> /var/log/cwm.log
```

This would run the script each day at 2:25 AM. Be sure to have an additional line break after the line.

3.20 Install the standard Models and Users

In case of Upgrade:

- Open Firefox 3 and browse to http://localhost/cwm/application
- Click on upgrade to upgrade your database

3.21 Connecting 2 CWM instances

Requirements: 2 CWM instances with at least version 0.9.9.2

Server: CWM instance that provides a *library model* to the client instance.

Client: CWM instance that references parts of the library model in it's *local models.*

Server configuration:

- nothing to do here

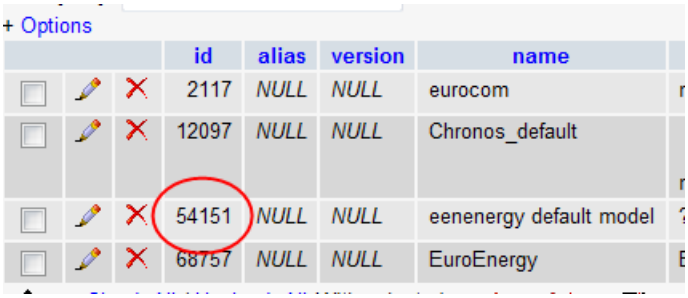
Client configuration:

- In *server.ini* add the following sections

```
[remoteserver]

cim = http://url.of.server.cwm/application/main.php

[remoteuser]

cim = cimuser
```

For each remote server add an entry in the *remoteserver* section, where the key (e.g. "lib" or "cim") is an arbitrary system wide unique name for the server and the value is the url of the remote CWM instance. To log into the remote application the appropriate user denoted in the *remoteuser* section will be used. The user must exist in both instances.

- Check with PHPmyadmin the database of the Server CWM.
    - Look in the table Model or Package for the ID.



- Log into the **backend application** of the CWM client.
- create a new Model instance.
- Set the *umi* property of the instance to an umi of a remote Model or Package, e.g.

    `urn:umi:cim:Model:54151`

    or

    `urn:umi:cim:Package:54151`

[umi notation: `urn:umi:`{remote server key}`:`{object id of referenced Model or Package}]

- Save the Model instance.

- Log into the CWM. The referenced Model is contained in the model tree.

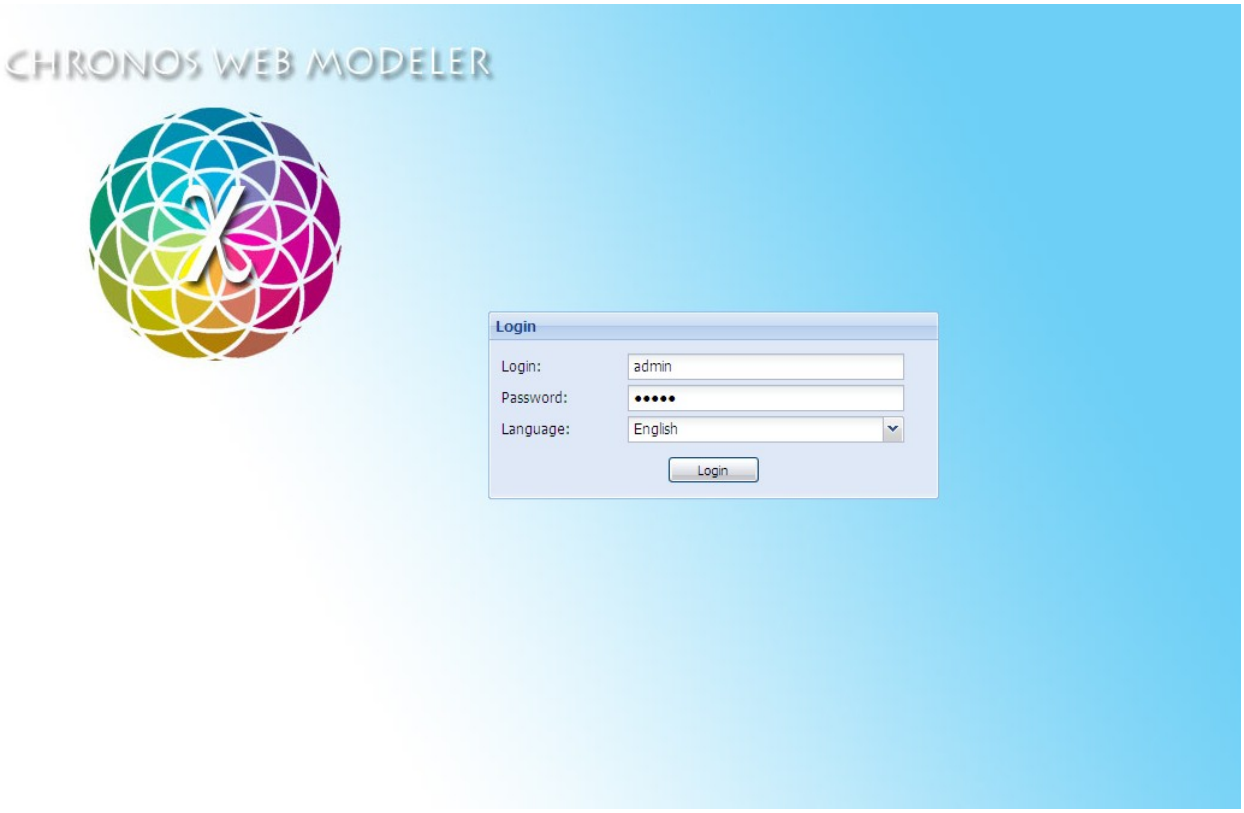
4. Get started

4.1 Administrate users and roles

- Click on start application to start the backend application
- Use the standard user:admin with the password:admin to login
- Click on `Administration` to manage users and roles
- Change immediately the password for the user `admin`
- Create so many users and roles as necessary.

4.2 Test the CWM

- Browse to http://localhost/cwm/gui/
- Login
- welcome to the CWM!!!



4.3 Test the CWB

NOTE the CWB is now integrated as start window of the CWM

- Browse to http://localhost/cwm/Browser
- Login
- select a model
- welcome to the CWB!!!