Documentation

MyCoRe – WCMS-Module

Authors:
Andreas Trappe
Michael Brendel

PDF created with pdfFactory trial version www.pdffactory.com

Toc:

Introduction 3
How does MyCoRe-WCMS work 4
  Architecture 4
    WCMS-module in general 4
    How does User interface builds a web page 5
    XSL framework 6
    Templates 7
Installation 8
  DocPortal 8
  Standalone 9
Configuration 11
  How to change the default template 11
  Manage access to WCMS with WCMSUserDB.xml 12
    Introduction 12
    User database of the WCMS 12
Usage 15

PDF created with pdfFactory trial version www.pdffactory.com

Introduction

Key functions:

- Add, Edit, Delete and translate web content
- Distributed Web-Browser bases administration
- Automatic generation of menu structures
- Templates
- Multi lingual
- Sitemap
- Logging, Backup
- "Plug and Play" installation

PDF created with pdfFactory trial version www.pdffactory.com

How does MyCoRe-WCMS work

Architecture

WCMS-module in general

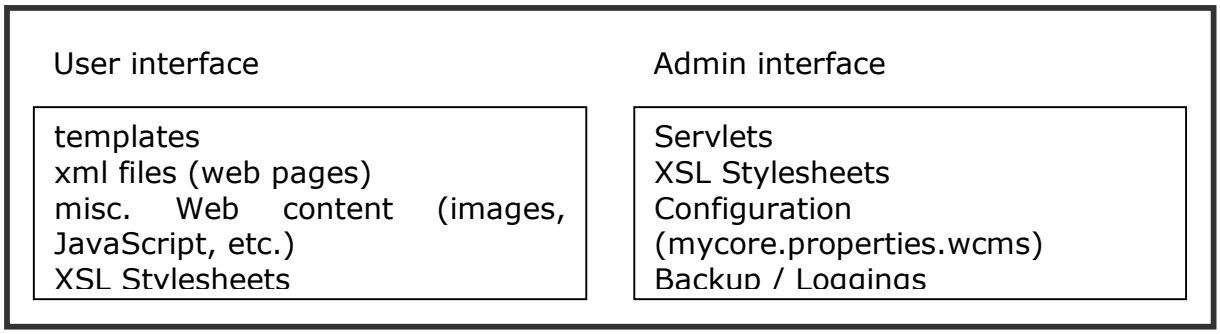

Abbildung 1 – WCMS module

The WCMS-Module in general consists of two main parts: User interface and admin interface.

User interface is responsible for generation and providing of web pages.
It uses XML-Navigation Base to create menus, and XML-files for displaying right Web-Pages and a couple of XSL-style sheets to build the framework of the Website (watch XSL framework).

Admin module is responsible for management of web pages displayed by the User interface.
It uses Servlets to read, write… web content and modifications on navigation base.

PDF created with pdfFactory trial version www.pdffactory.com

How does User interface builds a web page

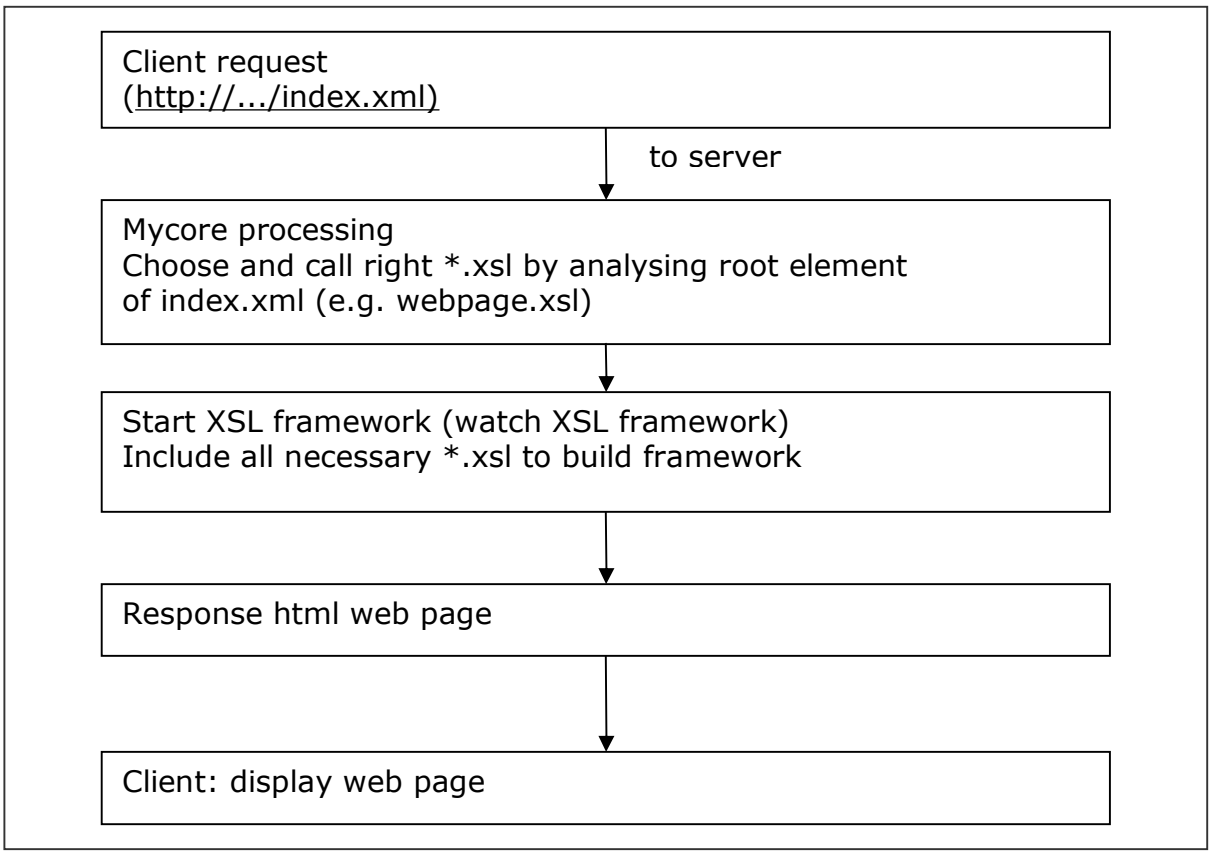

Abbildung 2 – Building a web page by User module

PDF created with pdfFactory trial version www.pdffactory.com

XSL framework

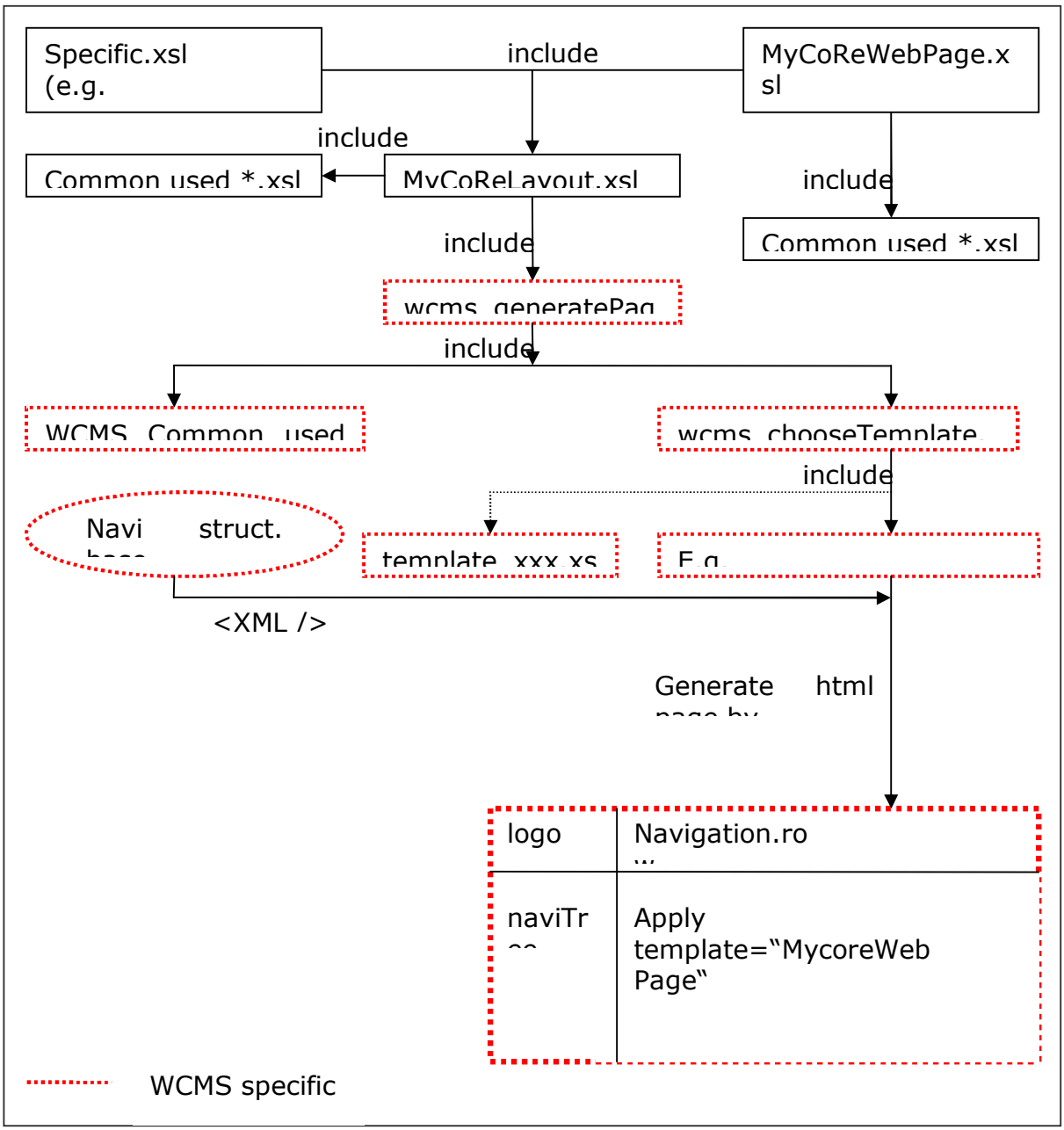

Abbildung 3 - XSL framework

PDF created with pdfFactory trial version www.pdffactory.com

Templates

Templates can understand as a method to change layout and site building within the same website.

PDF created with pdfFactory trial version www.pdffactory.com

Installation

The installation consists of two main parts.
1. At first the non generic parts of the WCMS will be copied to the target application (e.g. DocPortal).
2. Afterwards the application specific adaptations have to be done.
Thenceforward the installation will be automatically executed by running the mycoresample-application ant webapps.

DocPortal

To install MyCoRe-WCMS ensure the latest version of MyCoRe, Docportal, and MyCoReSample (www.mycore.de) is installed.

Copy module to application:

1. **Plugin Module**
Go to $MYCORESAMPLE_HOME/
ant wcms

Configuration:

2. **Point Default menu entries to your server**
open $MYCORESAMPLE_HOME/modules/module-wcms/uif/web/common/navigation.xml
replace "path_to_mycore_server" by DNS-Name of your MyCoRe-Server
E.g.: Your MyCoRe-Server is reachable at: "http://www.mycoreServer.de/mycore"
à replace "path_to_mycore_server" by "http://www.mycoreServer.de/mycore"
---
open $MYCORESAMPLE_HOME/modules/module-wcms/uif/web/index.html
replace "path_to_mycore_server" by DNS-Name of your MyCoRe-Server

3. **Adapt mycore.properties.wcms**
open $MYCORESAMPLE_HOME/modules/module-wcms/aif/config/mycore.properties.wcms
replace dummy paths by real ABSOLUTE paths of your Mycoresample-Application

Installation

4. **Install Module**
cd $MYCORESAMPLE_HOME
ant webapps

5. **Set file system permissions**
WCMS-Servlets must have the following permissions (the locations you have set in mycore.properties.wcms):
Read: "wcmsUserDBFile"
read + write: "content section" RECURSIVELY
read + write: "navigationFile"
read + write: "backupPath" RECURSIVELY
read + write: "logFile"

PDF created with pdfFactory trial version www.pdffactory.com

read + write: "footer"
read + write: "documentPath"
read + write: "imagePath"

6. **Restart Web-Application-Server**

Usage

Call website and log into WCMS (WCMS button right above). The default username is admin with password wcms.

Try to add, edit a page and assign different templates and take a look at the effect to your website...

Standalone

MyCoRe-WCMS as standalone application is a complete independent application.
It uses the mycore-core api to access common functions and own elements to realise the WCMS.

The thing you have to do generally is setup a document base for the WCMS application.
But, just follow the installation routine:

Generate application

1. **Set paths to applications**
Open $MYCORE_HOME/modules/module-wcms/appsSupport/standalone/build.xml
Reset values of: "env.MYCORE_HOME" and "env.MYCOREWCMS_HOME"

2. **Generate application**
Go to $MYCORE_HOME/modules/module-wcms
ant standalone

Configuration

3. **Point Default menu entries to your server**
open $MYCOREWCMS_HOME/modules/module-wcms/uif/web/common/navigation.xml
replace "path_to_mycore_server" by DNS-Name of your MyCoRe-Server
E.g.: Your MyCoRe-Server is reachable at: "http://www.mycoreServer.de/mycore"
à replace "path_to_mycore_server" by "http://www.mycoreServer.de/mycore"
---
open $MYCOREWCMS_HOME/modules/module-wcms/uif/web/index.html
replace "path_to_mycore_server" by DNS-Name of your MyCoRe-Server

4. **Adapt mycore.properties.wcms**
open $MYCOREWCMS_HOME/modules/module-wcms/aif/config/mycore.properties.wcms

PDF created with pdfFactory trial version www.pdffactory.com

replace dummy paths by real ABSOLUTE paths of your Mycore-WCMS-Application

5. **Adapt mycore.properties.***
open $MYCOREWCMS_HOME/config/mycore.properties.*
adapt entries

Installation

6. **Install Module**
cd $MYCOREWCMS_HOME
ant webapp

7. **Set file system permissions**
WCMS-Servlets must have the following permissions (the locations you have set in mycore.properties.wcms):
Read: "wcmsUserDBFile"
read + write: "content section" RECURSIVELY
read + write: "navigationFile"
read + write: "backupPath" RECURSIVELY
read + write: "logFile"
read + write: "footer"
read + write: "documentPath"
read + write: "imagePath"

8. **Restart Web-Application-Server**

Usage

Call website and log into WCMS (WCMS button right above). The default username is admin with password wcms.

Try to add, edit a page and assign different templates and take a look at the effect to your website...

PDF created with pdfFactory trial version www.pdffactory.com

Configuration

How to change the default template

1. Reset entry in navigation base
    a) open $navigationFile

    b) go to root tag <navigation ... />

c) replace value of attribute template="template_nameOfDefaultTemplateRightNow"
    by
    template="template_newNameOfDefaultTemplate"

2. Reset entry in chooseTemplate.xsl
a) open $MYCORESAMPLE_HOME/modules/module-wcms/uif/xsl/core/chooseTemplate.xsl

    b) replace in the area of "*set name of default template*" the following:
    <xsl:call-template name="template_nameOfDefaultTemplateRightNow">
      by
    <xsl:call-template name="template_newNameOfDefaultTemplate">

    c) cd $MYCORESAMPLE_HOME

    d) ant webapps

PDF created with pdfFactory trial version www.pdffactory.com

Manage access to WCMS with WCMSUserDB.xml

Introduction

The actual implementation of the WCMS doesn't allow to administrate any user data using a web oriented GUI. Thus, to add, delete or change users or user informations a system administrator has to configure the user database and possibly has to execute the userPassword encoder of the WCMS manually.

User database of the WCMS

The user database of the WCMS is a simple XML file called WCMSUserDB.xml. Each user got one entry in that database storing his usergroup he belongs to, his userID and userPassword and his real name. Furthermore a list of root nodes will be stored standing for elements of the navigation.xml under wich content can be changed.

There are 4 userClass definable:
- "systemAdmin" (do everything)
- "admin" (add, delete, edit, tranlate)
- "editor" (add, edit, translate)
- "author" (add, translate)

Location of the database

Normally the user database can be found at the following position:

$DOCPORTAL_HOME/modules/wcms/config/WCMSUserDB.xml

Structure of the database

As already mentioned, the user database is a simple xml file storing user informations as attributes of the user element. The root nodes, under wich the user can add and change content, will be stored as child elements of the user parent element.

Example 1: WCMSUserDB.xml

```
+-------------------------------------------------------------+
  <?xml version="1.0" encoding="ISO-8859-1"?>
    <userdb>
      <user userClass="systemAdmin"
```

PDF created with pdfFactory trial version www.pdffactory.com

```
        userID="sysAd"
        userPassword="ee7c6277441abd6d87367c3a7c5312ee"
        userRealName="Systemadministrator">
      <rootNode href="no">navi-main</rootNode>
      <rootNode href="no">navi-below</rootNode>
    </user>
    <user userClass="admin"
        userID="admin"
        userPassword="953320848fa1ed55c09031bc45bc9369"
        userRealName="Andreas Trappe">
      <rootNode href="no">navi-main</rootNode>
      <rootNode href="no">navi-below</rootNode>
    </user>
    <user userClass="editor"
        userID="micedit"
        userPassword="ebb5d7efc0472ac77534a55a7d1ffd"
        userRealName="Michael Brendel">
      <rootNode href="no">navi-main</rootNode>
      <rootNode href="no">navi-below</rootNode>
    </user>
    <user userClass="author"
        userID="author_ilm"
        userPassword="5a671c66aefea124cc08b76ea6d30bb"
        userRealName="Torsten Schmidt">
      <rootNode href="yes">
       /content/main/information/ilmenau.xml
      </rootNode>
      <rootNode href="yes">
       /content/below/ilmenau/diss_ilmenau.xml
      </rootNode>
    </user>
    <user userClass="author"
        userID="transengl"
        userPassword="583ce7497cda5ca007553ce902e"
        userRealName="Jutta Klein">
      <rootNode href="no">navi-main</rootNode>
      <rootNode href="no">navi-below</rootNode>
    </user>
  </userdb>
+-----------------------------------------------------------------+
```

Adding a new user to the database

To add a new WCMS user to the database you have to edit the WCMSUserDB.xml file. At first the new user should be classified ito an userClass.

After the classification you have to select a userID and a userPassword for the new user or the user has already given you a userID and userPassword he likes to use. Either way you have to encrypt the userPassword manually. This can be done by using the HashCipher.class of the WCMS. The HashCipher.class should be called like this:

PDF created with pdfFactory trial version www.pdffactory.com

CLASSPATH_of_WCMS:>java wcms.util.HashCipher Password

where CLASSPATH_of_WCMS is the JAVA CLASSPATH the Web application is looking for classes. Normally this should be:

$MYCORESAMPLE_HOME/webapp/WEB-INF/classes

<"String"> is the parameter wich will be encrypted. Copy and paste the output to the userPassword attribute of the new user. The last step to add a new user is to define his root nodes under wich he can add or modify content. A list of available root nodes can be found in the navigation.xml file. The navigation.xml file again can be found at the following position:

$MYCORESAMPLE_HOME/modules/wcms/common/navigation.xml

Valid root nodes are either complete menus like navi-main or navi-below (wich are elements of navigation.xml and mark the corresponding menu) or only parts of a menu. In the case of root nodes being only parts of a menu respectivly an item element in the navigation.xml file, the href attribute has to be used instead of the element name and the rootNode attribute "href" in the WCMSUserDB.xml has to be set to "yes".

PDF created with pdfFactory trial version www.pdffactory.com

Usage

PDF created with pdfFactory trial version www.pdffactory.com