



OpenCms Days 2011

Workshop Track:

***Upgrading from OpenCms 7.5 to
OpenCms 8***

Michael Emmerich,
Alkacon Software GmbH.



1. Migrating from OpenCms 7.x to OpenCms 8
 - Export and Import in OpenCms
 - Using the Update-Wizard
 - Required changes

2. Benefits of using OpenCms 8 with your OpenCms 7.x webapp
 - Available OpenCms 8 features without any modifications
 - OpenCms 8 feature that require modifications

3. Usage of OpenCms 7.x elements in OpenCms 8
 - Transformation of "old style" elements into OpenCms 8 usable elements
 - How to display xmlcontents as OpenCms 8 elements
 - Parallel usage of elements in OpenCms 7 and OpenCms 8 templates
 - Usage of old content list as OpenCms 8 elements



Good news:

Your existing OpenCms 7.x
webapp will run in OpenCms 8
– most likely without any
changes 😊



Migrating from OpenCms 7.x to OpenCms 8



Two possible ways to migrate

1. Exporting modules and content from OpenCms 7.x and import it into a new, fresh OpenCms 8
2. Use the OpenCms Update-Wizard on your existing installation



Manual export and import of modules and content will result in a clean system

- Only the required modules and content need to be exported
- Uses Module & Database management in OpenCms
- BUT: Historic versions of resources will be lost!



- Step 1:
 - Export all required modules from OpenCms 7.x
 - **Always export your modules from the root site!**
 - Result: One .zip file for each module

Administration View > Module Management

Module Management

Module actions

- New Module
- Import Module from Server
- Import Module with HTTP

Installed Modules

Installed Modules (35)

Module Group Search

Author info Resources Dependencies Resourcetypes Print Delete

E	A	D	Module [▲]	Title	Group	Version	
			de.baerenurlaub.frontend	BaerenUrlaub Frontend	Baerenurlaub	0.22	<input type="checkbox"/>
			org.opencms.editors	OpenCms 7 Common editors	OpenCms Editors	7.5.1	<input type="checkbox"/>
			org.opencms.editors.editarea	OpenCms 7 EditArea editor	OpenCms Editors	7.5.2	<input type="checkbox"/>
			org.opencms.editors.fckeditor	OpenCms 7 FCKeditor WYSIWYG editor plugin	OpenCms Editors	7.5.4	<input type="checkbox"/>
			org.opencms.frontend.layoutpage	OpenCms 7 frontend Layoutpage	OpenCms frontend	7.0.0	<input type="checkbox"/>
			org.opencms.frontend.photoalbum	OpenCms 7 Photo album frontend	OpenCms frontend	7.0.0	<input type="checkbox"/>
			org.opencms.frontend.templatetwo	OpenCms 7 Template Two frontend	Template Two	7.5.1	<input type="checkbox"/>
			org.opencms.frontend.templatetwo.demo	OpenCms 7 Template Two demo site	Template Two	7.5.2	<input type="checkbox"/>
			org.opencms.frontend.templatetwo.demo.content	OpenCms 7 Template Two demo site content	Template Two	7.5.3	<input type="checkbox"/>
			org.opencms.jquery	OpenCms 7 JQuery library	OpenCms frontend	7.5.1	<input type="checkbox"/>
			org.opencms.locale.de	OpenCms 7 German localization for the workplace	OpenCms Localization	7.5.3	<input type="checkbox"/>
			org.opencms.locale.es	OpenCms 7 Spanish localization for the workplace	OpenCms Localization	7.5.3	<input type="checkbox"/>
			org.opencms.welcome	OpenCms 7 Welcome page and release notes	OpenCms Release Notes	7.5.4	<input type="checkbox"/>
			org.opencms.workplace	OpenCms 7 Workplace base	OpenCms Workplace	7.5.4	<input type="checkbox"/>

Export



- Step 2:
 - Export the content from OpenCms 7.x
 - **Always export your content from the root site!**
 - Export content & accounts
 - Multiple exports on large OpenCms installations (per site/folder)
 - Result: One or more content export .zip files

Administration View > Database Management > Export Database

Export Database

Export target zip file

Select file: content

Export type

Include resource data:

Include account data:

Include project data:

Export settings

Include unchanged files:

Include system folder:

Resource changed since: (Click on the "New" button on the right side to activate this element)

Recursive:

Modified in Project:

Export as files:

Resources to export

Resource: /sites/travel/

Ok Cancel

- Step 3:
 - Import all required modules in OpenCms 8
 - **Always import your modules in the root site!**

- Module imports are published automatically
- Do not forget libs! (if you have some)

Administration View > Module Management
Module Management

Module actions

New Module Import Module from Server Import Module with HTTP

Installed Modules

Installed Modules (1 - 50 of 52) Author info Resources Dependencies Resourcetypes Print Delete

E	X	D	Module [▲]	Title	Group	Version	
			com.alkacon.opencms.v8.article	OpenCms 8 Article	OpenCms 8 types	1.0.9	
			com.alkacon.opencms.v8.event	OpenCms 8 Basic Event	OpenCms 8 types	1.0.11	
			com.alkacon.opencms.v8.list	OpenCms 8 List	OpenCms 8 types	1.0.4	
			com.alkacon.opencms.v8.login	OpenCms 8 Login	OpenCms 8 types	0.4	
			com.alkacon.opencms.v8.news	OpenCms 8 News	OpenCms 8 types	1.0.5	
			com.alkacon.opencms.v8.search	OpenCms 8 Search	OpenCms 8 types	1.0.6	
			com.alkacon.opencms.v8.twitter	OpenCms 8 Twitter	OpenCms 8 types	1.0.1	
			de.baerenurlaub.frontend	BaerenUrlaub Frontend	Baerenurlaub	0.22	
			org.opencms.ade.config	OpenCms 8 ADE Configuration	OpenCms ADE	8.0.0	
			org.opencms.ade.containerpage	OpenCms 8 ADE Containerpade Editor	OpenCms ADE	8.0.0	
			org.opencms.ade.editprovider	OpenCms 8 ADE toolbar direct edit provider	OpenCms ADE	8.0.0	
			org.opencms.ade.galleries	OpenCms 8 ADE Galleries	---	8.0.0	
			org.opencms.ade.publish	OpenCms 8 ADE	OpenCms ADE	8.0.0	
			org.opencms.ade.sitemap	OpenCms 8 Sitemap	OpenCms ADE	8.0.0	

Import

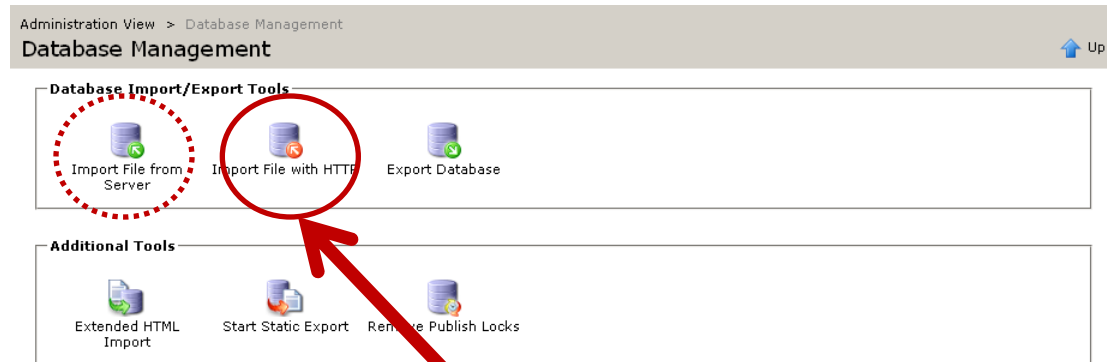


- Step 4:
 - Import all content exports in OpenCms 8.0
 - **Always import your content in the root site!**
 - Large imports should be done via the server directly, very large imports should be unpacked on server

– Content must be published after import

– Restart Servlet-Container

– **Done !**



Import

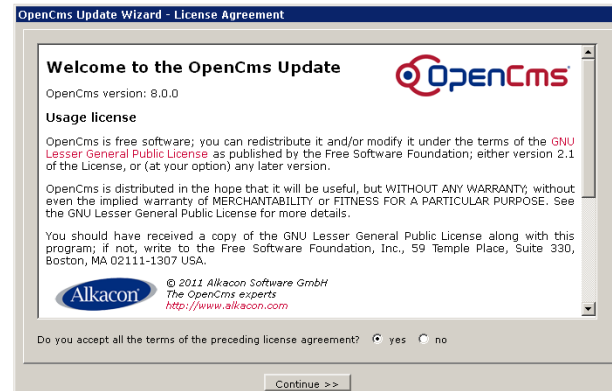
The Update Wizard automatically updates an existing OpenCms 7.x installation to OpenCms 8:

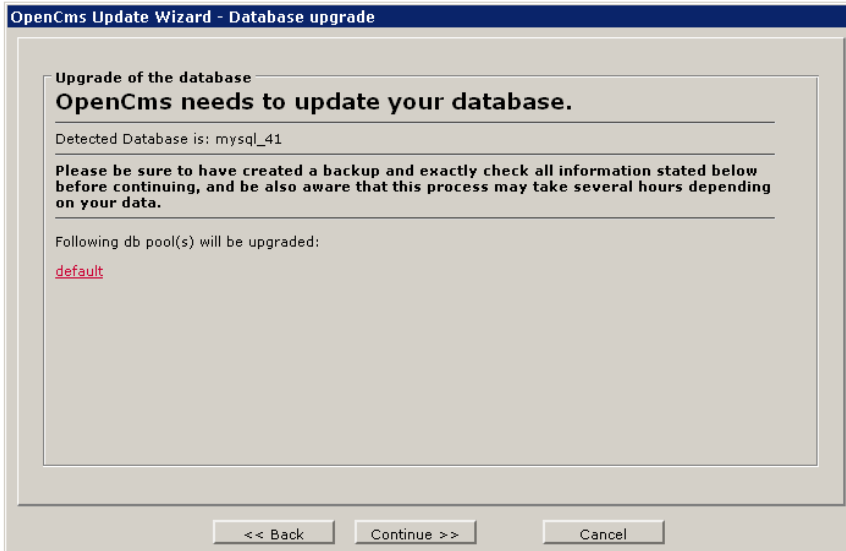
- Modification of the Database Structure
- Modification/Enhancement of the OpenCms configuration files
- Replacement of all Java libraries (where required)
- Update of all OpenCms core modules

- Existing content is untouched
- Historic versions remain



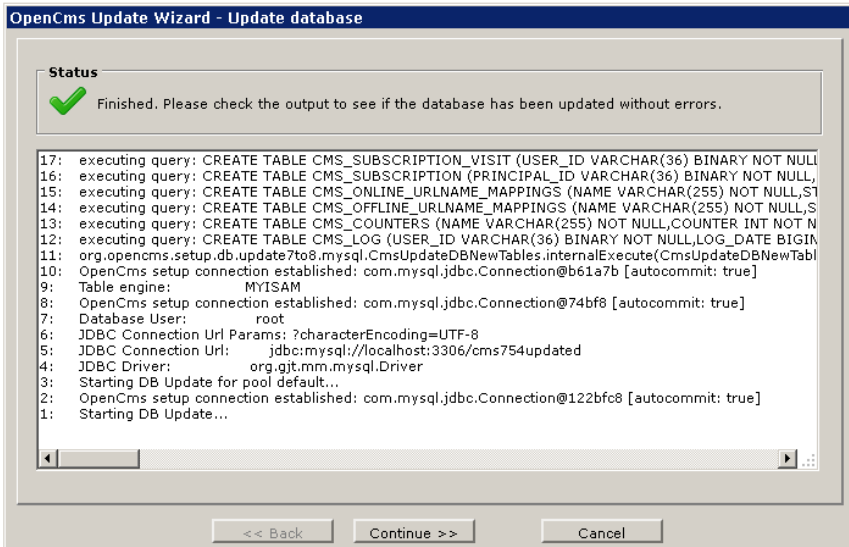
- Delete previous Update-Wizards!
- Deploy the Wizard
- Enable Wizard in `opencms.properties`:
[...]
`wizard.enabled=true`
[...]
- Start the Wizard with
`http://[ServerName]/[WebappName]/update`





- Step 1:

- Updates Database Structure
- New tables will be added



- Progress report
- Check for errors / warnings

OpenCms Update Wizard - Settings

OpenCms update settings
⚠ Make sure you have administration permissions to update your system.

Identification
Admin User: ?
Admin Password: ?

Settings
Update Site: ?

Installation Paths
OpenCms Application rootpath: C:\dev\Tomcat6.0\webapps\cms754updated\
OpenCms Configuration folder: C:\dev\Tomcat6.0\webapps\cms754updated\WEB-INF\config\

<< Back Continue >> Cancel

- Step 2:

- Enter Admin account
- Enter update site
(on multisite installation only one site required)

OpenCms Update Wizard - XML Configuration Files Update

XML changes available for update

opencms-workplace.xml

- [Add missing description for the new wizard to some explorer types](#)
- [Update open gallery link to use the advanced gallery](#)
- [Adds icon rules for various explorer types.](#)
- [Update the menu rules for various explorer types](#)
- [Change the default direct edit provider](#)
- [Update some context menu entries](#)

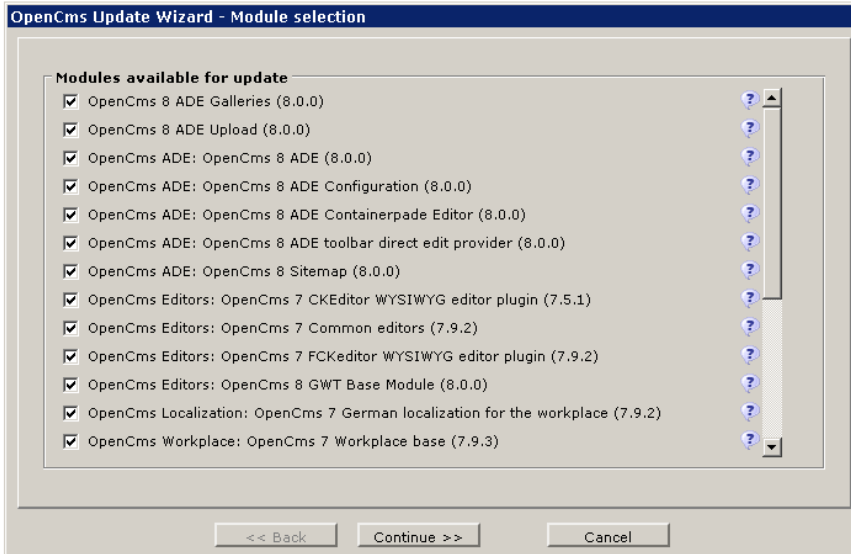
opencms-vfs.xml

- [Add new loader classes](#)
- [Add new loader parameters to JSP loader](#)
- [Add new parameters of resource type](#)
- [Add new widget definitions for ADE galleries](#)
- [Add new content collectors](#)

opencms-system.xml

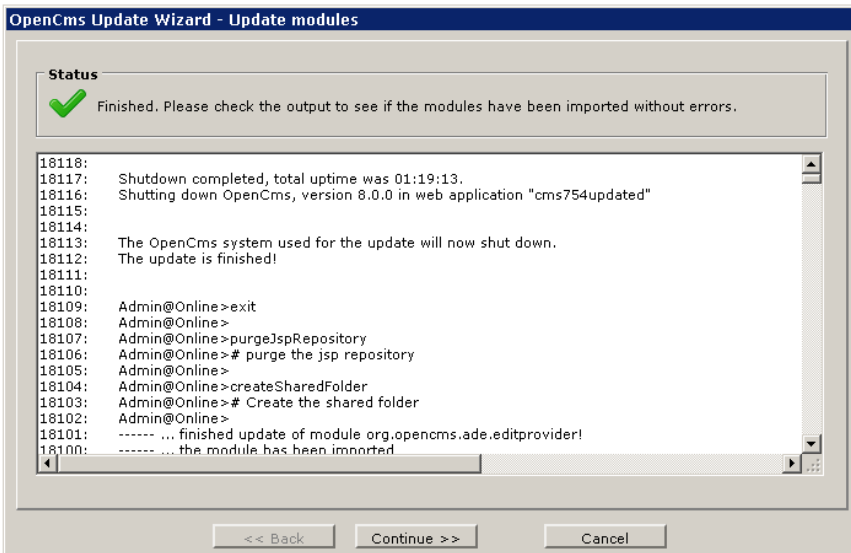
<< Back Continue >> Cancel

- Select configuration modifications
(Default: all)
- Do only change something here, if you know what you are doing....

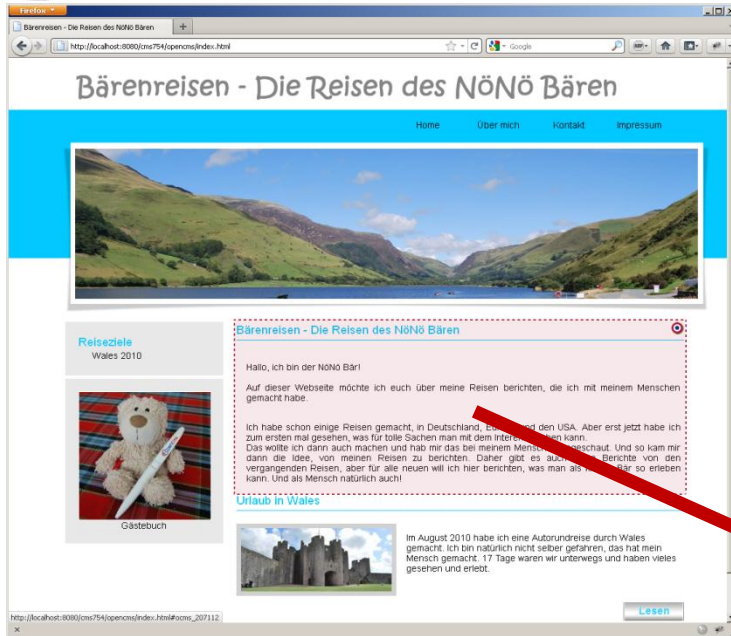


- Step 3:

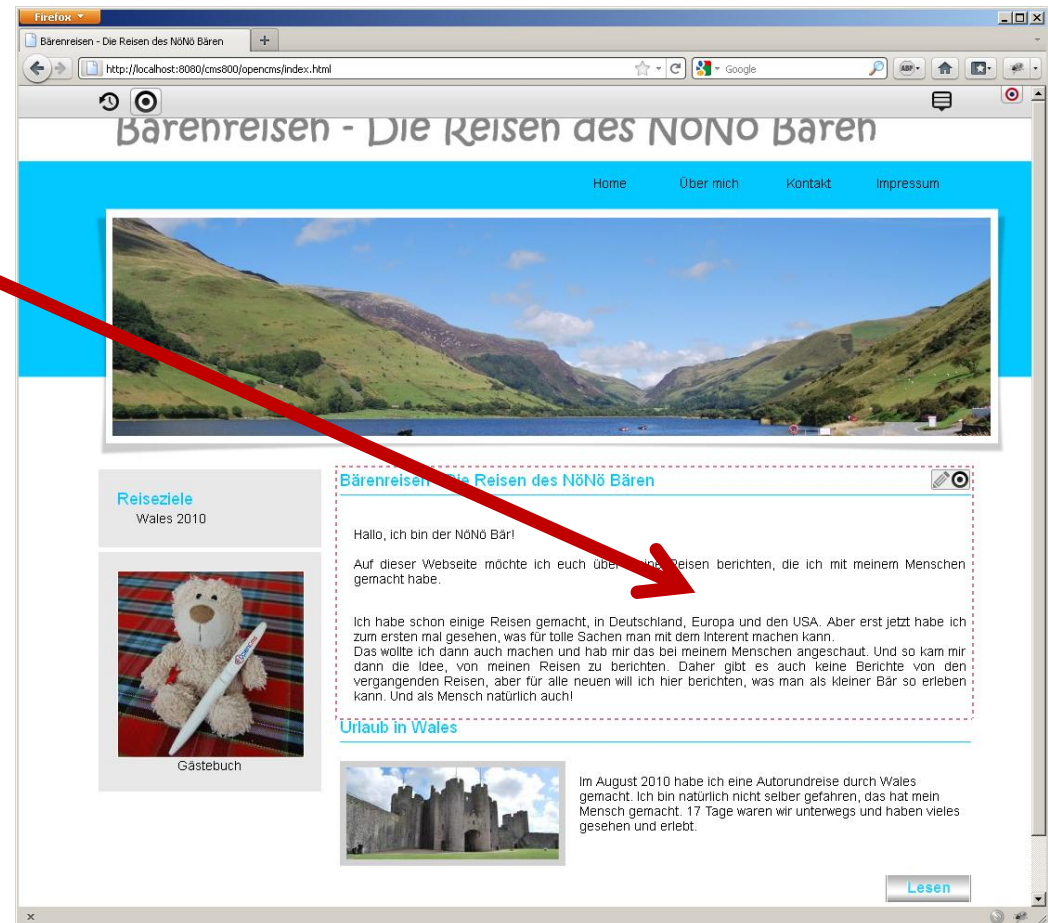
- Select modules
- Recommendation: Select all modules!



- Progress report
- Check update.log for errors or warnings
- Restart Servlet-Container
- **Done !**



From OpenCms 7.5.4....



... to OpenCms 8



Modifications that might be required:

- If you use `<cms>` Taglib and JSTL for your JSPs, no further modifications are required!
- It is possible that you have to modify your JavaBeans or scriplet code due to Core API changes
- BUT: Those changes are only minimal
 - Updating from OpenCms 6 to OpenCms 7 was much more complicated because of more API changes



Benefits of using OpenCms 8 with your OpenCms 7.x webapp



The existing OpenCms 7.x webapp will run in OpenCms 8 as before, and **uses** the new features in your templates without any changes :

- OpenCms 8 Direct Edit / Availability
- OpenCms 8 Publishing
- New Upload Dialog
- New Galleries
- Shared Site

You will benefit of all Core-Bugfixes and enhancements since OpenCms 7.x



The existing OpenCms 7.x webapp **does not** use:

- OpenCms 8 Drag & Drop
- OpenCms 8 Configuration
- OpenCms 8 Sitemap

It is required to modify your XSD, templates and elements. It also requires additional configuration files in the OpenCms VFS.



The existing OpenCms 7.x webapp **can** use the none ADE-enhancements:

- Content Subscription Engine
- Device-Tag for device-depending templates
- XSD-Choice function

This requires changes to your templates and elements.

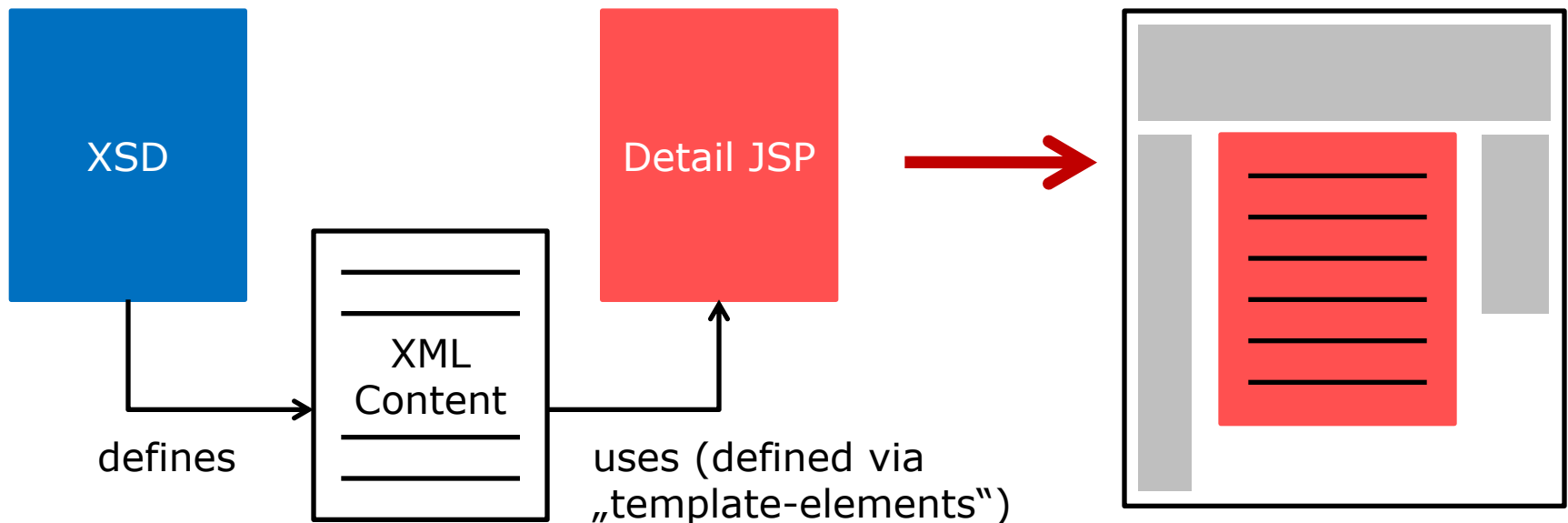


Usage of OpenCms 7.x elements in OpenCms 8



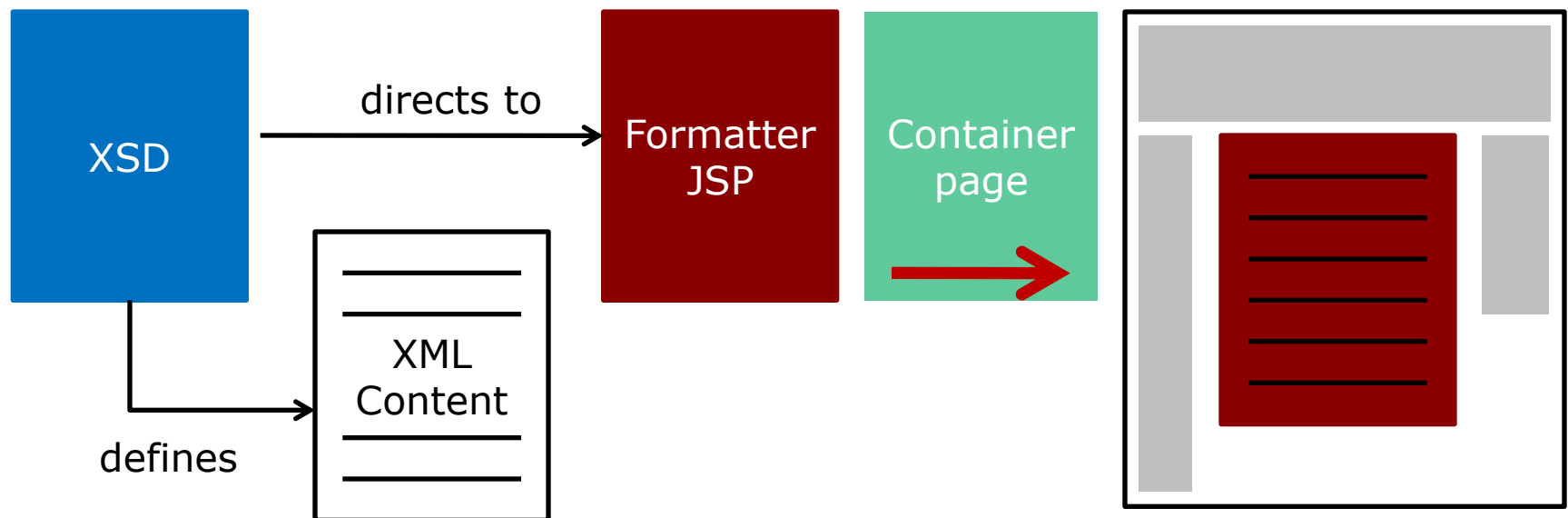
In OpenCms 7.x the following components were involved in displaying your (structured) content:

- XSD: Defines the structure of your content
- XML-content files: contain the content
- Detail-JSP: JSP to render your content



In OpenCms 8 this has changed:

- XSD: Defines the structure of your content and the formatter
- XML-content files: contain the content
- Formatter(s): JSPs to render your content



What must be done to transform your OpenCms 7.x contents and JSPs into drag&dropable elements in OpenCms 8?

The good answer:

- Only 3 easy steps are required!
- For „normal“ contents it only takes a few minutes to do the modifications



Step 1: XSD: Add additional annotation nodes:

- `<formatters>`: Add one or more formatters to display the content (mandatory)
- Optional, often used additions:
 - `<settings>`: Settings that can be accessed in the formatters and whose values are defined via the ADE
 - `<headincludes>`: Includes of .js and .css files that must be included in the HTML-Head, like custom styles, needed JQuery imports, etc.



Changes in XSD (<xsd:appinfo> node):

[...]

```
<formatters>
  <formatter minwidth="400" uri=path_to_formatter.jsp" />
  <formatter minwidth="200" uri=„path_to_other:formatter.jsp" />
</formatters>
```

```
<settings>
  <setting name="boxschema" nice-name="Box Schema" type="string"
    widget="select" widget-config="box_schema1:Schema 1|box_schema2:Schema
    2|box_schema3:Schema 3" />
</settings>
```

```
<headincludes>
  <headinclude type="javascript" uri=„path_to_js.js" />
  <headinclude type=„css" uri=„path_to_css.css" />
</headincludes>
```

[...]

Step 2: Detail JSP is modified into a formatter:

- Replacement of `<cms:contentload>` with `<cms:formatter>`
 - In most cases a simple replacement of the two tags
- Optional: Create different formatters for different containers (left, middle, center,...)
- Configurable OpenCms 7.x structured content can be transformed in multiple formatters:



Transformation Detail-JSP -> Formatter

OpenCms 7 (Detail-JSP)

```
[...]  
<cms:include property="template"  
  element="head" />  
  
<cms:contentload  
  collector="singleFile"  
  param="% (opencms.uri) "  
  editable="true">  
  
<cms:contentaccess var="content"  
  scope="page" />  
[...]  
${content.value.Title}  
[...]  
</cms:contentload>  
<cms:include property="template"  
  element="foot" />
```

OpenCms 8 (Formatter)

```
[...]  
<!-- no template-include -->  
  
<cms:formatter var="content"  
  val="value">  
  
<!-- no init of contentaccess -->  
[...]  
${content.value.Title}  
[...]  
</cms:formatter>  
  
<!-- no template-include -->
```

Detail-JSPs and formatter

- If you want to use your content „old style“ and with OpenCms 8 templates simultaneously:
 - Make a copy of your old detail pages and modify it as a formatter
 - „Old style“ will use „template-elements“ property and the old detail page
 - OpenCms 8 -Template will use formatter information from XSD and new formatter jsp
- Additional fine tuning (divs, styles etc.) might be required in your new formatters



Step 3: Registration of content types in the OpenCms 8 sitemap configuration

The screenshot shows a configuration window with the following fields and values:

- IsDefault:** (Click on the "New" button on the right side to activate this element)
- Source file:**
- Destination:**
 - Folder:**
 - Pattern:**
- Formatter:** (Click on the "New" button on the right side to activate this element)

- **Source File:** The model file for this content
 - You have to create one model file via the Workplace -> New function
- **Destination:** Where to store new content
 - Folder: The folder in the VFS to store the content files
 - Pattern: The naming patter for new content files
- **Formatter (optional):** Overwrites info in XSD

Usage of List-Elements

- List-Elements show a list of news, events, xmlcontent in general
- In OpenCms 7.x, we called it a „List-Jsp“ with code like this:

```
[...]  
<cms:contentload collector="allInSubTree" param="...">  
[...]  
<cms:contentaccess var="content" scope="page" />  
${content.value.Title}  
</cms:contentload>  
[...]
```


Content lists are now seen as content-types in OpenCms 8

- Define a new resource type for your list-element
 - Create the XSD
 - Transform the List-JSP into a formatter
 - Register the new defined contenttype

```
<cms:formatter>
<cms:contentload collector=„${collector}" param=„${colparam}">
[...]
<cms:contentaccess var="content" scope="page" />
${content.value.Title}
</cms:contentload>
</cms:formatter>
```

Optional: List configuration

- Selection of the collector / configuration of collector params
- Number of elements to be displayed
- Other options

- Make your list configurable within the new defined resourcetype via the content editor
- Make your list configurable with resourcetype settings in the `<settings>` node of the schema
- Combination of both is possible



Example of a configurable list (via the settings)

```
[...]  
<cms:formatter>  
  <c:set var=„collector"><cms:elementsetting name=„collector"  
    default=„allInSubTree" /></c:set>  
  <c:set var=„colparam"><cms:elementsetting  
    name=„colparam" /></c:set>  
  
  <cms:contentload collector=„${collector}" param=„${colparam}">  
    [...]  
    <cms:contentaccess var="content" scope="page" />  
    ${content.value.Title}  
  </cms:contentload>  
  
</cms:formatter>  
[...]
```

Summary



- Upgrading to OpenCms 8 is easy
- You can use your existing application like before
- It takes 3 easy Steps to use existing content in OpenCms 8 templates with all OpenCms 8 functions!



Our recommendation:

Update to OpenCms 8 now 😊



QUESTIONS ?

Fragen?

- Any Questions?

¿Preguntas?

Questiones?



Thank you very much for your attention

Michael Emmerich
Alkacon Software

<http://www.alkacon.com>

<http://www.opencms.org>

