OpenCms Days 2008

Technical Track:

*Secrets of using the OpenCms CRE*

Michael Moossen,
Alkacon Software GmbH
Agenda

- What is the OpenCms CRE?
- Workplace Improvements
  - Publishing
  - Deleting
  - XML Content editing
  - Categorization
- Frontend Improvements
  - Link management
  - The %(link) macro
- User defined relation types
- Implementation details
Concept: Problems

- Referencing:
  Moving files around will break links

\[
\begin{align*}
\text{<html>}
\text{<a href="b.html" >}
\text{....}
\text{</a>}
\text{</html>}
\end{align*}
\]

Rename `b.html` into `c.html`

\[
\begin{align*}
\text{<html>}
\text{<a href="b.html" >}
\text{....}
\text{</a>}
\text{</html>}
\end{align*}
\]

- Reference querying:
  Where is my resource referenced?
Concept: Technical Issues

- Referencing
  - Move Semantic
    - Copy + Delete (OpenCms 6)
  - Resource locking/publishing

- Reference querying
  - Read and parse content to get the references
Concept: Solution

- Referencing
  - Move semantic
    - Path change
    - To keep references with path and id
    - Lazy reference path update

- Reference querying
  - Specialized database tables with relation information

This is the OpenCms Content Relation Engine
Concept: Relation types

• Weak
  – Example: This is a link.
  – XML Page:
    • Html Tag <A>
  – XML Content:
    • Default for VfsFile / VarFile types

• Strong
  – Example: This is my dog: 🐶
  – XML Page:
    • Html Tags <IMG> and <OBJECT>
  – XML Content:
    • Annotation option for VfsFile / VarFile types
Workplace

- **Improvements**
  - Publish dialog
    - Publishing with related resources
    - Broken links check
  - XML Content Editor
  - Delete dialog
  - Link Validation Tool

- **New features**
  - Relations overview
  - Categorization
• Publishing with related resources
Workplace: Publish

- Broken link check during publishing
Workplace: Delete

- Broken link check during deletion
Workplace: Configuration

- Publish related resources
  - Default behavior
    - Related resources will be published
  - Configurable in opencms-workplace.xml
    - */default-preferences/workplace-preferences/
      workplace-generaloptions/publishrelatedresources

- Broken link check (publish/delete)
  - Default behavior
    - Referenced resources can only be deleted by users having the role VFS_MANAGER
  - Configurable in opencms-workplace.xml
    - */default-preferences/workplace-preferences/
      workplace-generaloptions/allowbrokenrelations
Workplace: Link validation

- Fast internal link validation
Workplace: Editor

- New XML schema types with link validation:
  - OpenCmsVfsFile:
    - VFS link
  - OpenCmsVarFile:
    - VFS or external link
Workplace: Relations menu

- New relations context menu
- Relations overview

<table>
<thead>
<tr>
<th>#</th>
<th>L</th>
<th>S</th>
<th>Relation</th>
<th>Name</th>
<th>Title</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>Html link</td>
<td>/demo_en/view/procedure/de/index.html</td>
<td>Herstellungsverfahren</td>
<td>layoutpage</td>
<td>16479</td>
</tr>
</tbody>
</table>
Workplace: Categories

- Centralized category definition
## Workplace: Categories

- Categories dialog

### Category Assignment

<table>
<thead>
<tr>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path: /demo_en/dictionary/haggis.html</td>
</tr>
<tr>
<td>Title: Haggis</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Assigned categories</th>
<th>Available categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assigned categories (2)</td>
<td>Available categories (3)</td>
</tr>
<tr>
<td>Path</td>
<td>Description</td>
</tr>
<tr>
<td>Search</td>
<td>Search</td>
</tr>
<tr>
<td>A</td>
<td>Category</td>
</tr>
<tr>
<td>Wurstsorten</td>
<td>Brühwurst / Pre-cooked</td>
</tr>
<tr>
<td>Kochwurst / Cooked</td>
<td>Rohwurst / Raw</td>
</tr>
</tbody>
</table>
**Workplace: Categories**

- **Category widget**

![Image of the category widget]

**Haggis** is a traditional Scottish dish. It is traditionally served with the Burns supper on the week of January 25, when Scotland's national poet, Robert Burns, is commemorated. He wrote the poem *Ode Tae a Haggis*, which starts "Fair fa' your honest, sonsie face, Great chieftain o' the puddin'-race!"

There are many recipes, most of which have in common the following ingredients: sheep's 'pluck' (heart, liver and lungs), minced with onion, oatmeal, suet, spices, and salt, mixed with stock, and traditionally boiled in the animal's stomach for approximately three hours.

Haggis somewhat resembles stuffed intestines (pig intestines otherwise known as chitterlings or the kokoretsi of traditional Greek cuisine), sausages and savoury puddings of which it is among the largest types. As the 2001 English edition of the Larousse Gastronomique puts it, "Although its description is not immediately appealing, haggis has an excellent nutty texture and delicious savoury flavour." (p592)

Most modern commercial haggis outside Scotland is prepared in a casing rather than an
Frontend

- Categories
- Link management
  - XML Pages
  - XML Contents
- The %(link) macro
- Permalinks
Frontend: Categories

- Categories Collector:
  org.opencms.files.collectors.
  CmsCategoryResourceCollector
    - Name: allKeyValuePairFiltered
    - Parameter: categoryTypes

- Categories API:
  org.opencms.relations.
  CmsCategoryService
    - readCategoryResources
    - readResourceCategories
    - addResourceToCategory
    - removeResourceFromCategory
Frontend: XML Pages

- Move semantic:
  - Links stored with path and id
- Relation types:
  - Automatic, hard coded
- Broken link handling:
  - No broken link handling
Frontend: XML Content

- **Move semantic:**
  - New XML schema types with link validation
    - OpenCmsVfsFile
    - OpenCmsVarFile
  - Links stored with path and id

- **Relation types:**
  - Default is weak
  - Optional annotation setting

- **Broken link handling:**
  - Default is to invalidate node
  - Optional annotation setting
  - NOTE: Node has to be optional
• Nested schema:

```xml
<xsd:complexType name="OpenCmsTTImage">
    <xsd:sequence>
        <xsd:element name="Image" type="OpenCmsVfsFile" />
        <xsd:element name="Description" type="OpenCmsString" />
    </xsd:sequence>
</xsd:complexType>

<xsd:annotation>
    <xsd:appinfo>
        <layouts>
            <layout element="Image" widget="ImageGalleryWidget" />
        </layouts>
    </xsd:appinfo>
</xsd:annotation>
```
Frontend: XML Content

- Outer schema

```xml
<xsd:complexType name="OpenCmsTTArticle">
  <xsd:sequence>
    <xsd:element name="Title" type="OpenCmsString" />
    <xsd:element name="Paragraphs" type="OpenCmsTTParagraph" />
    <xsd:element name="Config" type="OpenCmsConfig" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

<xsd:annotation>
  <xsd:appinfo>
    <relations>
      <relation
        element="Paragraphs/Image/Image"
        invalidate="parent"
        type="strong" />
    </relations>
  </xsd:appinfo>
</xsd:annotation>
```
Frontend: XML Content

- **Original**

  **Extranet of the Wurst**

  ![Salami image]

  In our Extranet you will find secret Wurst recipes and other great stuff. You need to login in order to access the Extranet.

  To the login page of the Wurst Extranet

- **Invalidate false**

  **Extranet of the Wurst**

  ![Salami image]

  In our Extranet you will find secret Wurst recipes and other great stuff. You need to login in order to access the Extranet.

  To the login page of the Wurst Extranet

- **Invalidate true/node**

  **Extranet of the Wurst**

  ![Salami image]

  In our Extranet you will find secret Wurst recipes and other great stuff. You need to login in order to access the Extranet.

  To the login page of the Wurst Extranet

- **Invalidate parent**

  **Extranet of the Wurst**

  ![Salami image]

  In our Extranet you will find secret Wurst recipes and other great stuff. You need to login in order to access the Extranet.

  To the login page of the Wurst Extranet
Frontend: %(link) macro

- JSP Pages
  - Move semantic
    - %(link.weak) macro
  - Include dependencies
    - %(link.strong) macro
  - Usage:

```xml
After saving:
<image src="<cms:link>%(link.strong:images/load.gif)</cms:link>" />
```

- After moving target:

```xml
<image src="<cms:link>%(link.strong:images/reload.gif:fe055c44-120a-11dd-8a3f-111d34530985)</cms:link>" />
```
Frontend: Permalink

- Permalink:
  - Resource access by path
    /index.htm
  - Resource access by id
    /permalink/87972a79-12be-11dd-a2ad-111d34530985.htm
  - Exact link is shown in most explorer dialogs
User defined relation types

• How to define a relation?
  – From a XML Page: `<A>`, `<IMG>`, etc.
  – From a XML Content: `VfsFile` and `VarFile`
  – From a PDF File:
    • User defined relation types

• Configuration in `opencms-vfs.xml`

```
<relationtypes>
    <relationtype name="pdfweak" type="WEAK" />
    <relationtype name="pdfstrong" type="STRONG" />
</relationtypes>
```
User defined relation types

- **CmsObject**
  - To add a relation
    - `addRelationToResource`
  - To remove relations
    - `deleteRelationsFromResource`
  - To read relations
    - `getRelationsForResource`

- **CmsRelationFilter**
  - Relation direction
    - `SOURCE`, `TARGET`, `BOTH`
  - Resource
    - `filterResource`
  - Additional options
    - Type, subresources, etc.
Implementation details

- XML Page
- XML Content
- Database tables
Release Notes for OpenCms 7.0.4

Please contact the core development team on the opencms-dev mailing list in case you are interested in supporting the development of OpenCms.

Thank you for using OpenCms and have fun with OpenCms 7.0.4 :-)

<?xml version="1.0" encoding="UTF-8"?>

<TTImages xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="opencms://system/modules/org.opencms.frontend.templatetwo.demo/schemas/nested/image.xsd">
  <TTImage language="de">
    <Image>
      <link type="WEAK">
        <target><![CDATA[/sites/default/test.jpg]]></target>
        <uuid>9cb2ffbb-dbcc-11dc-8613-9f7a6a6ce285</uuid>
      </link>
    </Image>
    <Description>My nice picture</Description>
  </TTImage>
</TTImages>
### Implementation: Database

- Offline & Online tables

<table>
<thead>
<tr>
<th>CMS_OFFLINERESOURCE_RELATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESOURCE_SOURCE_ID</td>
</tr>
<tr>
<td>RESOURCE_SOURCE_PATH</td>
</tr>
<tr>
<td>RESOURCE_TARGET_ID</td>
</tr>
<tr>
<td>RESOURCE_TARGET_PATH</td>
</tr>
<tr>
<td>RELATION_TYPE</td>
</tr>
</tbody>
</table>

- Null UUID for broken relations
Thank you very much for your attention

Michael Moossen
Alkacon Software GmbH

http://www.alkacon.com
http://www.opencms.org