OpenCms Days 2011

Workshop Track:
Creating Mobile Templates with OpenCms 8
Rüdiger Kurz
Alkacon Software GmbH
Agenda

- Device dependent content presentation
- End-user demo
- The `<cms:device>`-tag
- Development strategies for mobile templates
- Developer demo
- OpenCms-FlexCache directives
- Implementation details
Device dependent presentation

- **Motivation:**
  Increase the availability of web content on different devices:
  - Desktop based Browser (e.g. Firefox on Linux)
  - Mobile Phone (e.g. iPhone or Android Phone)
  - Other devices

- **Earlier Solution:**
  Create one template for each device.

- **Drawback:**
  Multiple templates must be maintained, the user may have to use different URLs for each device.

- **OpenCms 8:**
  Present the content device dependent in one single template, without knowledge of an extra URL!
The `<cms:device>`-tag

- For the device detection, the OpenCms tag library has been enhanced with an additional tag called `<cms:device>`

- The tag has only one attribute: `type` specifying the type of the output device(s).

  `<cms:device type="mobile">`

- The idea is to define sections in JSP templates that are only displayed if the user’s device is in the option list of the given device types.
The `<cms:device>`-tag

```html
<%@ taglib prefix="cms" uri="http://www.opencms.org/taglib/cms" %>

<html>
  <head>
    <title>Simple Device Demo</title>
  </head>
  <body>
    <cms:device type="mobile">
      <p>You are browsing with a mobile device!</p>
    </cms:device>
    <cms:device type="desktop">
      <p>You are browsing with a desktop device!</p>
    </cms:device>
    <cms:device type="desktop, console">
      <p>You are browsing with a console or a desktop!</p>
    </cms:device>
    <p>Other content displayed on all devices…</p>
  </body>
</html>
```
Development strategies

• With the help of the `<cms:device>`-tag you are able to develop a template that:
  – Selects different style-sheets (CSS)
  – Switches between other templates
  – Contains the code for all devices

• Disadvantage for selecting different CSS
  – Complex to maintain, because the same CSS classes have to be defined twice.
  – Data transfer overhead, because all the content will be transferred to the client even content that is not displayed ➔ Long waiting time for connections with small bandwidth.
# Development strategies

<table>
<thead>
<tr>
<th></th>
<th>All-in-one Template</th>
<th>Switching Template</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Maintaining</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>LOC</td>
<td>+ (less)</td>
<td>- (more)</td>
</tr>
<tr>
<td>Reusability</td>
<td>==</td>
<td>==</td>
</tr>
<tr>
<td>Caching</td>
<td>==</td>
<td>==</td>
</tr>
<tr>
<td>Readability</td>
<td>-</td>
<td>+</td>
</tr>
</tbody>
</table>
FlexCache directives

- A JSP that uses the `<cms:device>`-tag should be cached for each device output.
- OpenCms supports to cache different variations of HTML-output for the device-tag.
- The template developer can simply set the property “cache” to the value “device” on JSPs that use the `<cms:device>`-tag.
Implementation details

• For the `<cms:device>`-tag a methodology to detect the current user’s device dynamically is needed.

• Possible device detection methodologies:
  
  – Do nothing to detect different devices
  – Use JavaScript
  – Use CSS `@media` handheld
  – Use server-side code PHP, JSP, ASP, etc.
  – Use existing 3rd party libraries e.g. WURFL
Implementation details

- Alkacon decided to use a simple server-side implementation for detecting the devices.

- The device detection is based on the HTTP request header information:
  - HTTP-Accept
  - User-Agent

- Supported device types:
  - desktop : desktop computers
  - mobile : mobile devices
  - console : game consoles

- In a future version of OpenCms the tag could also consider the clients resolution for a better detection.
QUESTIONS?

¿Preguntas?

Questiones?
Thank you very much for your attention

Rüdiger Kurz
Alkacon Software

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