

HowToLanguageSwitching

Language Switching in Multilingual Publications

Some Background

Setting up languages in a Lenya publication

HTTP Content Negotiation

Language Switching Widgets

Lenya will start a session for each user and associate the language chosen with that session if it is not the default language. As long as the user is using the default language, all links inside the application are untouched and will contain just the document id, for example:

```
http://www.server.com/lenya/switchlanguage/authoring/subpage.html
```

Only if the user chooses to use a different language, all links in the page are automatically rewritten to point to specific language versions of the pages. If for example the user switches to Italian while *englisch* is the default language, the same link will read:

```
http://www.server.com/lenya/switchlanguage/authoring/subpage_it.html
```

Note that *different language* refers to the default language of the publication, not the preferred language sent my the user's browser.

Links pointing to a different language version

Switching the language is as easy as linking to a document with a different target language. Lenya will recognise that the user is requesting a different language than the current one associated with the user's session and replace the browsing language with the new one. That's all.

Global Language Switching Widgets

If you want to place the famous "language flags" in a fixed place to allow the user to switch to a different language at any given point in time, you can put these controls into the XSL template. By using some variables, you make sure that these links will always be generated correctly.

A modified `page2xhtml.xsl` might read like this:

```
...
<div id="page">
<table width="100%" cellpadding="0" cellspacing="0" border="0">
  <tr>
    <td id="publication-title">Welcome to the Default Publication!</td>
    <td id="project-logo"></td>
  </tr>
</table>

<table width="100%" cellpadding="0" cellspacing="0" border="0">
  <tr>
    <td>
      <a href="{ $root }{ $document-id }_de.html">DE</a>
    </td>
    <td>
      <a href="{ $root }{ $document-id }_en.html">EN</a>
    </td>
    <td>
      <a href="{ $root }{ $document-id }_it.html">IT</a>
    </td>
  </tr>
</table>

<xsl:apply-templates select="xhtml:div[@id = 'tabs']"/>
...
```

Note the `<a>` tags:

```
<a href="{ $root } { $document-id } _de.html" >DE</a>
```

That's all. It's that easy!

If you want to implement generic language links, you can use the `publication-languages-csv` attribute of the page envelope module:

```
<map:parameter name="languages" value="{page-envelope:publication-languages-csv}" />
```

It returns a comma-separated list of the publication languages (`de,en,it`) which can be transformed to a set of links using a recursive template. Don't forget to disable the link to the current language version.