

Selector

One of the [CocoonConditionals](#)

A selector allows simple control logic to be embedded inside a [Sitemap](#). Logic can be a simple if-then-else structure, or a switch statement. They are similar to Matcher in the tests they can perform, but are more flexible: matchers simply define whether a pipeline is evaluated or not (essentially a simple if statement). Selectors can drive processing down multiple routes.

There are several selectors available. These can perform similar operations to the equivalent matchers.

Declaration

Selectors are declared in the sitemap, as follows:

```
<map:selectors default="browser">
  <map:selector name="load"
    src="org.apache.cocoon.selection.MachineLoadSelector">
    ...
  </map:selector>

  <map:selector name="user"
    src="org.apache.cocoon.selection.AuthenticationSelector">
    ...
  </map:selector>

  <map:selector name="browser"
    src="org.apache.cocoon.selection.BrowserSelector">
    <browser name="explorer" useragent="MSIE"/>
    <browser name="lynx" useragent="Lynx"/>
    <browser name="mozilla5" useragent="Mozilla/5"/>
    <browser name="mozilla5" useragent="Netscape6"/>
    <browser name="netscape" useragent="Mozilla"/>
    ...
  </map:selector>
</map:selectors>
```

Note the default attribute which defines the default selector for pipelines that don't explicitly define one.

Usage

Example usage (note similarity with xsl:choose):

```
...
<map:match pattern="*">
  <map:generate type="serverpages" src="test/{1}.xsp"/>

  <map:select type="browser">
    <!-- you could insert parameters here as well -->
    <map:when test="explorer">
      <map:transform src="stylesheets/w3c-2-msie.xsl"/>
    </map:when>
    <map:when test="lynx">
      <map:transform src="stylesheets/dynamic-page2html-text.xsl"/>
      <map:serialize/>
    </map:when>
    <map:when test="netscape">
      <map:transform src="stylesheets/ns4.xsl"/>
    </map:when>
    <map:otherwise>
      <map:transform src="stylesheets/w3c.xsl"/>
    </map:otherwise>
  </map:select>

  <map:transform src="stylesheets/dynamic-page2html.xsl"/>
  <map:serialize/>
</map:match>
...
```

Available Implementations

- Browser Selector – tests User-Agent HTTP header, to allow browser recognition
- Code Selector – uses a snippet of Java code to do the test
- [HostSelector](#) – tests the Host header in the request
- [ParameterSelector](#) – matches string against an internal sitemap parameter ??
- [HeaderSelector](#) – matches string against the request headers.
- [RequestParameterSelector](#) – matches string against the request parameters.
- Session Selector – matches string against a session parameter
- [SimpleSelector](#) – matches on string literals
- Cookie Selector – matches string against a cookie value
- See also [RequestParameterModule](#)