

# RegexURIDefaultsMatcher Tutorial

Bernhard Huber, [berni\\_huber@a1.net](mailto:berni_huber@a1.net)

## 1 RegexURIDefaultsMatcher Tutorial

This document describes how-to stream line your sitemap when you are using the *Paginator* transformer.

## 2 Paginator Transformer

The *Paginator* transformer emits URIs of the form `foo({N}).html` for this site. `{N}` represents the current paged URI page.

Thus the sitemap has to handle now URIs of the form `foo.html`, and `foo({N}).html`. Using a simple redirect for non paginated URIs may be an option. The solution described here tries to handle both URI formats uniquely by using *RegexURIDefaultsMatcher*.

## 3 Streamlining The Sitemap

The sitemap of this site uses the same processing for URI of the form `name.html`, and `name(N).html`.

Thus it would be nice if a matcher matches both URIs, and provide the optional page value encoded as `(N)` to the *Paginator* transformer.

## 4 Sitemap Reconstructed

The most difficult task in using *RegexURIDefaultsMatcher* is finding the correct regexp.

This site wants to match URI of the form `*(*).html`. Thus the regexp is something like `[a-z]+\([0-9]+\)`, making the `(N)` optional yields `[a-z]+\([0-9]+\)?`, finally adding the fragments of interests yields `([a-z]+\([0-9]+\))?`.

Setting the default fragment values is simple, just define that fragment 2 becomes by default `(1)`, and fragment 3 becomes 1. Setting fragment 2 may be not necessary, it is just done for completeness reason.

Thus finally the sitemap is changed to

```
<map:match
  pattern="^(^[^(/)0-9]+)(\(([0-9]+)\))?.html$"
  type="regexp-defaults">
  <map:parameter name="map-default:2" value="(1)"/>
  <map:parameter name="map-default:3" value="1"/>
  ...
  <map:transform type="paginate"
    src="pagesheets/wmlpages.xml">
    <map:parameter name="page" value="{3}"/>
  </map:transform>
  ...
```

This sitemap fragment handles both paginated, and non paginated URI requests. The optional ( {N} ) encoded page value is passed to the *Paginator* as {3} sitemap parameter value.

## 5 Summary

This article describes one use case of *RegexURIDefaultsMatcher*, streamlining the sitemap in case of using *Paginator* transformer.

Finding the correct regexp might be the biggest obstacle for using this matcher.