

# LinkStatus Generator (2.1 legacy document)

## Table of contents

1 Comments.....3



**Table of contents**

1 LinkStatus Generator.....3



**Warning:**

This document was copied as is from the Cocoon 2.1 documentation, but has not yet been fully reviewed or moved to its new home.

## 1. LinkStatus Generator

The LinkStatus Generator emits a list of links that are reachable.

The LinkStatusGenerator has several configuration options.

**include-name**

RE pattern for including links

By default include-name is empty.

**exclude-name**

RE pattern for excluding links.

By default exclude-name is defined as `.*\.(gif|png|jpe?g|js|css)(\?.*)?$`.

**link-content-type**

expected MIME type of xml document requested on view *link-query-view*

By default link-content-type is set to `application/x-cocoon-links`.

**link-view-query**

A query-string appended to the crawling URL

By default link-view-query is set to `cocoon-view=links`.

**user-agent**

HTTP user-agent for requesting links, By default user-agent is set to value of `org.apache.cocoon.Constants.COMPLETE_NAME`, ie. Apache Cocoon 2.1-dev

**accept**

Not currently used

A simple example might help to use the LinkStatusGenerator effectively:

Add the LinkStatusGenerator to the components in your `sitemap.xml`

```
... <map:components> ... <map:generators default="file"> ... <map:generator
name="linkStatus" src="org.apache.cocoon.generation.LinkStatusGenerator"/>
</map:generators> <map:serialize default="html"> <map:serializer name="links"
src="org.apache.cocoon.serialization.LinkSerializer"/> </map:serialize> </map:components>
<map:views> <map:view> <map:view from-position="last" name="links"> <map:serialize
type="links"/> </map:view> ... </map:view>
```

Next define in your pipeline to use the LinkStatusGenerator

```
<map:match pattern="/linkStatus"> <map:generate type="linkStatus" name="my-root"/> ...
<map:serialize/> </map:match>
```

## 1. Comments

add your comments