

# XML Serializer (2.1 legacy document)

## Table of contents

1 Comments.....6

## Table of contents

1 XML Serializer.....	3
1.1 Sitemap Configuration.....	3
1.1.1 Configuration Example.....	3
1.2 Pipeline Usage.....	5
1.3 Further Reading.....	5

**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. XML Serializer

The xml serializer is the simplest possible serializer. It generates an xml document from the sax events. This serializer is used for serializing to any XML document format, ie. SVG, WML, VRML, et. al.

- Name : xml
- Class: org.apache.cocoon.serialization.XMLSerializer
- Cacheable: yes.

### 1.1. Sitemap Configuration

The XML Serializer is declared in the sitemap serializers section.

```
<map:serializers ... .. <map:serializer name="xml"
src="org.apache.cocoon.serialization.XMLSerializer" mime-type="text/xml"
logger="sitemap.serializer.xml" pool-max="32"> <!-- serializer configurations --> ...
</map:serializer> ...
```

XML Serializer can be configured, specifying elements inside of the <map:serializer> body.

#### 1.1.1. Configuration Example

The following XML Serializer snippet is setting encoding configuration for the XML Serializer

```
<map:serializer name="xml" src="org.apache.cocoon.serialization.XMLSerializer"
mime-type="text/xml"> <encoding>ISO-8859-1</encoding> </map:serializer>
```

This configuration will result in xml output of the form

```
<?xml version="1.0" encoding="ISO-8859-1"?> ...
```

The XML Serializer accepts following configuration parameters. These configurations are not Xalan specific.

Name	Xalan Default Value	Comment
cdata-section-elements	none	cdata-section-elements specifies a whitespace delimited list of the names of elements whose text node children should be output using CDATA sections. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a> .
doctype-public	none	doctype-public specifies the public identifier to be used in the document type declaration.
doctype-system	none	doctype-system specifies the system identifier to be used in the document type declaration. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>

encoding	none	encoding specifies the preferred character encoding that the Transformer should use to encode sequences of characters as sequences of bytes. The value of the attribute should be treated case-insensitively. The value must only contain characters in the range #x21 to #x7E (i.e., printable ASCII characters). The value should either be a charset registered with the Internet Assigned Numbers Authority <a href="#">[IANA]</a> , <a href="#">[RFC2278]</a> or start with X-. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>
indent	yes	A Flag for toggling indent. This flag toggles only if some elements should trigger a line break.
media-type		media-type specifies the media type (MIME content type) of the data that results from outputting the result tree. The charset parameter should not be specified explicitly; instead, when the top-level media type is text, a charset parameter should be added according to the character encoding actually used by the output method. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>
method		The method attribute identifies the overall method that should be used for outputting the result tree. Other non-namespaced values may be used, such as "xhtml", but, if accepted, the handling of such values is implementation defined. If any of the method values are not accepted and are not namespace qualified, then {@link javax.xml.transform.Transformer#setOutputProperty} or {@link javax.xml.transform.Transformer#setOutputProperties} will throw a {@link java.lang.IllegalArgumentException}. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>

omit-xml-declaration		omit-xml-declaration specifies whether the XSLT processor should output an XML declaration; the value must be yes or no. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>
standalone		standalone specifies whether the Transformer should output a standalone document declaration; the value must be yes or no. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>
version		version specifies the version of the output method. When the output method is "xml", the version value specifies the version of XML to be used for outputting the result tree. The default value for the xml output method is 1.0. When the output method is "html", the version value indicates the version of the HTML. The default value for the xml output method is 4.0, which specifies that the result should be output as HTML conforming to the HTML 4.0 Recommendation [HTML]. If the output method is "text", the version property is ignored. See <a href="#">section 16 of the XSL Transformations (XSLT) W3C Recommendation</a>

Former property buffer-size is deprecated, and is ignored.

The XML Serializer sets the method property to xml.

## 1.2. Pipeline Usage

Using the XML Serializer in a pipeline is just setting the serializer type to xml. The following code snippet uses the XML Serializer:

```
... <map:match pattern="*.xml"> <map:generate... ... <map:serialize type="xml"/> ...
```

## 1.3. Further Reading

The XML serializer is usable for serializing any SAX events to a plain xml output. The various xml documents requires in most cases a proper configuration of following parameters

- doctype-public
- doctype-system

Moreover the mime-type, and name attribute of the serializer definition shall be set properly.

Read the XML serializer configuration user documentation for [SVG/XML](#), and [WML](#) in order to

understand using the XML serialiazer for serializing to some specific XML content type.

## **1. Comments**

add your comments