

OutputEncoding

This [Sitemap](#) excerpt shows how to configure the encoding on documents generated by the HTML and XML [Serializers](#):

```
<!-- these definitions go into the map:serializers element -->

<!-- configure the XML serializer to use iso-8859-1 encoding -->
<map:serializer
  name="xml"
  mime-type="text/xml; charset=iso-8859-1"
  src="org.apache.cocoon.serialization.XMLSerializer"
  pool-max="32"
  pool-min="16"
  pool-grow="4"
>
  <encoding>iso-8859-1</encoding>
</map:serializer>

<!-- configure the HTML serializer to use iso-8859-1 encoding -->
<map:serializer
  name="html"
  mime-type="text/html"
  src="org.apache.cocoon.serialization.HTMLSerializer"
>
  <encoding>iso-8859-1</encoding>
</map:serializer>
<!-- configure the XML serializer to supply "text/html" -->
<map:serializer name="html"
  mime-type="text/html; charset=utf-8"
  logger="sitemap.serializer.html"
  pool-grow="2" pool-max="64" pool-min="2"
  src="org.apache.cocoon.serialization.XMLSerializer">
  <doctype-public>-//W3C//DTD XHTML 1.0 Strict//EN</doctype-public>
  <doctype-system>http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd</doctype-system>
  <!-- No XML declaration to force MS-InternetExplorer into standards compliant mode -->
  <omit-xml-declaration>yes</omit-xml-declaration>
  <omit-namespaces>yes</omit-namespaces>
  <encoding>UTF-8</encoding>
  <indent>yes</indent>
</map:serializer>
```

Well, I'm sure *that* example works just fine, since ISO-8859-1 seems to be the default anyway! But my attempts to persuade Cocoon (via jetty) to label its output as UTF-8 have all been in vain. Just to clarify, Cocoon is correctly generating UTF-8 encoded characters, but something is slapping

```
Content-Type: text/html; charset=ISO-8859-1
```

in the HTTP headers. Needless to say, this combination induces browser indigestion. Any clues? – [TimGoodwin](#)