

NISO JATS 1.0 Preview Stylesheets: Quick Start Guide

Journal Information
Journal ID: N/A
ISSN: N/A

Article/Issue Information
Publication date: 2012

NISO JATS 1.0 Preview Stylesheets: Quick Start Guide

Mulberry Technologies, Inc.

Introduction

These stylesheets provide preview display of journal articles tagged using NISO JATS version 1.0 (or any of several related tag sets). How you run them depends upon several things:

1. How citations are tagged in your input. Are they provided with explicit punctuation and formatting? Or are they unformatted?
2. What kind of output you want: HTML (with CSS) or PDF? If your citations are unformatted, do you want the citations in the output to have APA-style punctuation or NLM/PubMed Central-style punctuation?
3. What kind of computing environment you want to run the stylesheets in: You can run these stylesheets with a standalone XSLT processor; with an XProc pipeline engine; or with Saxon PE or EE version 9.2 or later.

This document provides quick instructions on using the stylesheets in the most common ways, omitting some details for simplicity. It assumes that your input is in one of the following versions of the NLM / JATS vocabulary:

- NISO JATS (Journal article tag set) Publishing or Authoring, version 1.0.
- NLM/NCBI Journal Publishing or Authoring, version 3.0.
- NLM/NCBI Journal Publishing or Authoring, version 2.3.
- An older variant of the NLM/NCBI DTD (your mileage may vary).

It also assumes you know how to invoke your XSLT or XProc processor on particular input and put the output where you want it to go.

For the full story on the limitations of these stylesheets and the assumptions they make about the input, see the *Users' Guide*.

For technical details, including a more precise description of technical dependencies, alternative ways to deploy and apply the stylesheets, and how to extend and modify them, see the *Technical Documentation*. That document also includes a more precise specification of the formatted and unformatted citation formats supported by this package.

Running the stylesheet

Standalone XSLT users

Some of the stylesheets in this distribution require XSLT 2.0, but most are in XSLT 1.0 and will work in any conformant XSLT processor. When running an XSLT 1.0 stylesheet with an XSLT 2.0 processor, it may give you warnings. Have no concern: They are designed to work correctly with either an XSLT 1.0 or XSLT 2.0 processor.

If your input has unformatted citations, you will need to perform citation pre-processing. If your citations are formatted, you can skip this step.

Note also that in the latter case (when no pre-processing is required), you can use an XSLT 1.0 processor in your web browser to perform the transformation and display the results dynamically. This scenario is described below as Example 1.

Citation pre-processing

- Produce a temporary document with citations formatted in NLM / Pub Med Central style by running `xslt/citations-prep/jats-PMCcit.xsl`. This is an XSLT 1.0 stylesheet.
- Alternatively, you may use a similar stylesheet for APA style, `xslt/citations-prep/jats-APAcit.xsl`. This is an XSLT 2.0 stylesheet and requires a 2.0 processor.

Then, on the temporary file you just created, or on your source file if your citations are already formatted:

For HTML output:

1. Run `xslt/main/jats-html.xsl` on your input (or on the temporary file, if you made one).
2. Copy `jats-preview.css` to the file system next to the result file(s).

For PDF output:

1. Run `xslt/main/jats-xslfo.xsl` on your input (or on the temporary file, if you made one).
2. Run an XSL-FO formatter on the XSLT output. These stylesheets have been tested using AntennaHouse formatter, but they should work in any conformant XSL-FO implementation.

XProc users

Pipelines of stylesheets may also be run using an XProc engine that supports XSLT 2.0, such as XML Calabash. (See the *Users' Guide* and the *Technical Documentation* for more pipelines for specialized applications.)

Run the appropriate pipeline from the following list:

- **For HTML output with PMC-style citations:** run `shells/xproc/jats-PMCcit-html.xpl`.
- **For HTML output with APA-style citations:** run `shells/xproc/jats-APAcit-html.xpl`.
- **For PDF output with PMC-style citations:** run `shells/xproc/jats-PMCcit-xslfo.xpl`.
- **For PDF output with APA-style citations:** run `shells/xproc/jats-APAcit-xslfo.xpl`.

If the input citations are already formatted, either the PMC or the APA pipelines can be run.

Saxon PE/EE users

Special "wrapper" stylesheets (or "shells") are provided for the convenience of Saxon users; these wrappers call other stylesheets in the distribution in pipelines in order to run pre- or post-processes along with the main transformation. Since they use extended XSLT functionality in a syntax defined by Saxon, they require Saxon PE or EE, version 9.2 or later. (Some older versions of Saxon also support the functionality.)

In addition to those listed here, other wrapper stylesheets are given in the distribution to demonstrate more specialized applications; see the *Users' Guide* and the *Technical Documentation* for more details.

Run the appropriate pipeline from the following list:

- **For HTML output with PMC-style citations:** run `shells/saxon/jats-PMCcit-html.xsl`.
- **For HTML output with APA-style citations:** run `shells/saxon/jats-APAcit-html.xsl`.
- **For PDF output with PMC-style citations:** run `shells/saxon/jats-PMCcit-xslfo.xsl`.
- **For PDF output with APA-style citations:** run `shells/saxon/jats-APAcit-xslfo.xsl`.

If the input citations are already formatted, either the PMC or the APA wrapper stylesheets can be run.

Examples

Example 1: HTML for display in a browser

1. The user has no unformatted citations, and wishes to open the XML document to preview directly in a web browser (such as Firefox, Safari, Chrome, Opera or Internet Explorer)
2. The appropriate stylesheet to use is `main/jats-html.xsl`. The user copies this stylesheet to the directory containing the XML source file, along with the CSS stylesheet to be applied to the result, `jats-preview.css`
3. The user edits the XML file, inserting a processing instruction at the top of the file as follows:

```
<?xml-stylesheet type="text/xsl" href="jats-html.xsl"?>
```

4. The user opens the XML file in the browser.

Example 2: Saxon from the command line

1. Again, HTML output is desired. The user has access to Saxon 9.4 PE, which supports Saxon extension functions.
2. The user determines that the data includes unformatted citations (using the `element-citation` element), and wishes them to be formatted according to APA guidelines.
3. The appropriate stylesheet to use is `shells/saxon/jats-APAcit-html.xsl`.
4. The user invokes Saxon to apply `shells/saxon/jats-APAcit-html.xsl` to the source document or documents (as described in Saxon documentation at <http://www.saxonica.com/documentation/using-xsl/commandline.xml>) generating a result file or set of files in HTML format
5. The user copies `jats-preview.css` into the subdirectory containing the results
6. HTML file output is now ready to be opened in a browser

Example 3: An XProc pipeline followed by XSL-FO formatting to produce PDF

1. This time, PDF output is desired. The user does not have Saxon, but is able to download and install XML Calabash, an open-source XProc engine supporting XSLT 2.0 (available at <http://xmlcalabash.com/>).

2. The user determines that the data includes unformatted citations (using the `element-citation` element), and wishes them to be formatted according to NLM/PMC guidelines.
3. The appropriate pipeline file is `shells/xproc/jats-PMCcit-xslfo.xpl`.
4. Following the Calabash documentation, the user generates a result file or set of files in XSL-FO format.
5. The XSL-FO file may now be opened and processed by an XSL-FO engine such as AntennaHouse.