

NIMAS Files Best Practices

By AEM Center at CAST

Published: 2021

The contents of this document were developed under a cooperative agreement with the US Department of Education, #H327Z190004. However, those contents do not necessarily represent the policy of the US Department of Education and you should not assume endorsement by the Federal Government. Project Officer, Rebecca Sheffield, Ph.D.



This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International license. <http://creativecommons.org/licenses/by-sa/4.0/>



National Center on Accessible Educational Materials (2011). *NIMAS Files Best Practices*. Wakefield, MA: National Center on Accessible Educational Materials

Table of Contents

1. General best practices for NIMAS/DAISY XML files.....	2
2. Page numbers.....	2
3. Multiple-page elements	3
4. Tables of contents	4
5. Glossary	5
6. Write-on lines (WOLs) or fill-in-the-blank lines	6
7. Resources for more information	6

1. General best practices for NIMAS/DAISY XML files

NIMAS XML files are required to use the Baseline Element Set found in the [NIMAS Technical Specification](#) and files must validate to the [DAISY DTD](#) (DAISY 2005-3). However, this is not the only requirement for valid NIMAS. Content in the print book must be tagged in accordance with the [DAISY Structure Guidelines](#). It is recommended that all publishers and vendors review the DAISY Structure Guidelines and refer to this guidance as needed when preparing NIMAS files.

The NIMAC provides a Validation Wizard as a stand-alone Windows program that provides DTD validation for NIMAS file sets. It also ensures that the XML is well-formed and that all referenced images are present in the images folder. NIMAC requires that all NIMAS files be validated using the tool before they are submitted to the NIMAC. The tool is available as a free download from the NIMAC system to all publishers and conversion vendors. The [NIMAC](#) also provides detailed instructions on required metadata for NIMAC validation, and staff are also available by phone and email to respond to questions.

2. Page numbers

Regardless of where they appear in the source material, page numbers should be marked up using the <pagenum> element at the beginning of each page. The “page” attribute should be used to indicate the type of page: front, normal or special. Placing

the `<pagenum>` element at the beginning of each page is of especial benefit for Braille transcription, and it does not hinder rendering the actual numbers elsewhere on page or screen. All page numbers should be included, whether or not they appear in print. Pages without numbers should be marked up with page numbers even if these will not be shown or rendered.

Occasionally source material will not have page numbers of any kind. Examples of instructional materials that may be created without printed page numbers include blackline masters, transparencies and worksheets. Yet the use of `<pagenum>` is necessary to distinguish one physical page from another. Unnumbered pages should be tagged but no page number should be included within the tag, allowing navigation by "next" and "previous" page controls.

Examples:

Unnumbered blackline master page: `<pagenum page="special"></pagenum>`

Unnumbered flashcard: `<pagenum page="special"></pagenum>`

3. Multiple-page elements

When a table spans more than one page of a printed work, it is recommended to put all page numbers in (separate) `<pagenum>`s prior to the start of the `<table>`; so a table that spanned three pages would have three `<pagenum>`s and then an opening `<table>` tag. Producers are also advised to add a `<prodnote>` that explains the page span, and a good location for this would be below `<pagenum>`(s) and above `<table>`.

Example:

```
<pagenum id="p009" page="normal">9</pagenum>
```

```
<pagenum id="p010" page="normal">10</pagenum>
```

```
<prodnote id="002.001.T04" render="optional">Please note that  
the following table spans two pages of the print source  
work.</prodnote>
```

```
<table>
```

4. Tables of contents

A table of contents (TOC) should be created as part of the NIMAS 1.1 source file for each print work that contains a TOC. The table of contents should be considered a nested list and each item in the table of contents should be marked as a list item. If page numbers are provided for the items listed, each item and each page number may be marked as separate "list item components" using the lic tag. The class attribute on lic specifies what type of list item component each one is (for example "entry" or "pagenum" in the provided example).

Example:

```
<level2 class="contents">

    <pagenum id="pii" page="front">ii</pagenum>

    <h2 class="contents">Table of Contents</h2>

    <list type="pl">

        <hd>The Cold War</hd>

        <li><lic class="entry">

            <a href="#page_1">Overview</a>

        </lic>

        <lic class="pagenum">

            <a href="#page_1">1</a>

        </lic>

    </li>

    ...

</list>

</level>
```

5. Glossary

The glossary terms and their definitions should follow the markup used for definition lists, as illustrated in the following example:

```
<dl>

    <dt>Accessibility</dt>

    <dd>The design of products, services and environments so
that all people, including people with disabilities, can use
them.</dd>

    <dt>Accessible Educational Materials</dt>

    <dd>Print- and technology-based educational materials,
including printed and electronic textbooks and related core
materials that are designed or converted in a way that makes
them usable across the widest range of individual variability
regardless of format</dd>

</dl>
```

If the glossary is subdivided alphabetically, a lower level than the one used for the glossary itself can be used to indicate the beginning of each section:

```
<level1 class="glossary">

    <pagenum id="page_831" page="normal">831</pagenum>

    <h1>Glossary</h1>

<level2 class="alphabetic_division ">

    <h2>A</h2>

    <dl>

        <dt>Accessibility</dt>

        <dd>Providing access for all people to web environments,
including people with disabilities.</dd>
```

6. Write-on lines (WOLs) or fill-in-the-blank lines

Currently there is not a specific element for write-on lines (WOLs) or fill-in-the-blank lines in the NIMAS 1.1 element set nor in DAISY2005. That being the case, it is acceptable to use a continuous series of underscore characters for this purpose. To match the level of granularity at which WOLs occur, it is recommended to code the characters as shown:

```
<span class="WOL">_____</span>      or      <span  
class="blankline">_____</span>
```

7. Resources for more information

The [AEM website](#) provides a wide variety of information on all things NIMAS-related, including a [NIMAS Technical Specification](#) page.

[Creating NIMAS Files](#) contains practical information and details about creating NIMAS XML files.

Go to the [NIMAS Exemplars](#) page to see examples of NIMAS-conformant files that include appropriate package files within a NIMAS-conformant fileset.

The [NIMAC](#) web site is the portal for the national repository of NIMAS filesets.

Review the [NIMAC Metadata Guidelines](#) for information on the required metadata that has to be included in a NIMAS package file.

The [DAISY Structure Guidelines](#) provides information on creating XML files valid to DAISY 2005.