DITA 1.3 Feature Article:
DITA 1.3 from A-Z

Author: Keith Schengili-Roberts, JoAnn Hackos
On behalf of the DITA Adoption Technical Committee

Date: 5 June 2015

This is a Non-Standards Track Work Product and is not subject to the patent provisions of the OASIS IPR Policy.
Notices

OASIS (Organization for the Advancement of Structured Information Standards) is a not-for-profit, international consortium that drives the development, convergence, and adoption of e-business standards. Members themselves set the OASIS technical agenda, using a lightweight, open process expressly designed to promote industry consensus and unite disparate efforts. The consortium produces open standards for Web services, security, e-business, and standardization efforts in the public sector and for application-specific markets. OASIS was founded in 1993. More information can be found on the OASIS website at http://www.oasis-open.org.

The OASIS DITA Adoption Technical Committee members collaborate to provide expertise and resources to educate the marketplace on the value of the DITA OASIS standard. By raising awareness of the benefits offered by DITA, the DITA Adoption Technical Committee expects the demand for, and availability of, DITA conforming products and services to increase, resulting in a greater choice of tools and platforms and an expanded DITA community of users, suppliers, and consultants.

DISCLAIMER: All examples presented in this article were produced using one or more tools chosen at the author's discretion and in no way reflect endorsement of the tools by the OASIS DITA Adoption Technical Committee.

This white paper was produced and approved by the OASIS DITA Adoption Technical Committee as a Committee Draft. It has not been reviewed and/or approved by the OASIS membership at-large.

Copyright © 2015 OASIS. All rights reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website. This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English. The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns. This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Document History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Draft</td>
<td>3 Mar 2015</td>
<td>Keith Schengili-Roberts</td>
<td>Initial draft</td>
</tr>
<tr>
<td>Second Draft</td>
<td>17 Mar 2015</td>
<td>JoAnn Hackos</td>
<td>Copyedit pass</td>
</tr>
<tr>
<td>Third Draft</td>
<td>5 Jun 2015</td>
<td>Keith Schengili-Roberts</td>
<td>Update against latest DITA 1.3 draft</td>
</tr>
<tr>
<td>Fourth Draft</td>
<td>15 Jun 2015</td>
<td>Keith Schengili-Roberts</td>
<td>Edits based on JoAnn Hackos' follow-up comments.</td>
</tr>
</tbody>
</table>

Last revision 5 June 2015

This is a Non-Standards Track Work Product and is not subject to the patent provisions of the OASIS IPR Policy.
| Revision   | Date       | Author              | Summary                                                        |
|------------|------------|---------------------|                                                               |
| Fifth Draft| 9 Jul 2015 | Keith Schengili-Roberts | Edits and content added based on additional comments.         |
| Sixth Draft| 20 Jul 2015| Keith Schengili-Roberts | Edits based on Stan Doherty's additional comments.             |
Table of Contents

DITA 1.3 from A-Z ................................................................................................................................. 5
  Introduction .............................................................................................................................................. 5
  Accessibility Markup for Tables ........................................................................................................... 5
  Branch Filtering ...................................................................................................................................... 5
  Cascade Attribute ................................................................................................................................... 6
  Content-Sensitive Help ............................................................................................................................ 6
  deliveryTarget (Goodbye print, Hello deliveryTarget) ............................................................................. 6
  Filtering Attributes Can Now Be Grouped ............................................................................................... 6
  Grouping Things Using div ..................................................................................................................... 7
  Highlight Domain ...................................................................................................................................... 7
  Inline Elements for Describing Markup Language ............................................................................... 7
  Key scope ................................................................................................................................................ 7
  Learning and Training Specialization ...................................................................................................... 8
  MathML and SVG Now Natively Supported ............................................................................................. 8
  RELAX NG ............................................................................................................................................... 9
  Release Management ............................................................................................................................... 9
  Same Topic References ............................................................................................................................ 9
  Sorting element ......................................................................................................................................... 9
  Table element—Turning Things Around ............................................................................................... 9
  Task for Troubleshooting ....................................................................................................................... 10
  Troubleshooting Note Type ................................................................................................................... 10
  Troubleshooting Topic ............................................................................................................................ 10
  Other Updates to Individual DITA 1.3 Elements and Attributes ............................................................ 10
The following sections describe the significant additions and enhancements appearing in the forthcoming DITA 1.3 specification. They are designed to provide a broad overview of the scope of the changes to come rather than providing in-depth information on each subject.

The information in this article is based upon the latest draft version of the All-Inclusive Edition of the draft DITA 1.3 specification, which contains all of the OASIS-approved specializations for DITA, including the Learning & Training specialization. In a working DITA 1.3 environment, you may not need or have access to all of the domains whose elements are described here. For example, if your firm does not use the Learning & Training specialization, the elements described in this article belonging to that domain may not be available for use. Similarly, an Information Architect can use constraints to restrict other DITA 1.3 elements from use.

The DITA 1.3 specification document is intended primarily for technology implementers, such as developers at software firms building tools that will work with the standard. The rest of us—Content Creators, Information Architects, Technical Documentation Managers, etc.—need a plain-language overview of what DITA 1.3 offers in terms of new features to expect and to plan for. This article is intended for the latter group.

The information in this article is based on the latest draft material from OASIS and could change prior to its official release, currently on-track for either the end of 2015 or early 2016.

Accessibility Markup for Tables

For those using Braille or spoken-word browsers, table-based content can be hard to understand, as it may not be easy to keep track of which cell value within a table row belongs to which column. DITA 1.3 includes attributes to identify the column name prior to a cell being read out. For example, a row with three cells that might otherwise be read aloud by a browser as “123, 7:00, 7:30”, could now be spoken with more context as: “Flight 123, Boarding 7:00, Expected Departure 7:30”.

Branch Filtering

Currently DITAVAL allows content creators to set output conditions for their content. The new DITA 1.3 element <ditavalref> provides the means to set unique conditions on a <topicref> (and its children, if any) within a map, allowing for branched publications, and can operate at multiple levels with more than one behavior. It also works at the root map level along with any child elements. <ditavalref> extends topic-level reuse by allowing the contents of an individual topic multiple times within a map, in each instance using different filtering conditions to produce different output.

For example, if your topics have information listed in metric and imperial units of measure, you can now easily generate tailored publications from a single map instead of having two separate maps or having to publish both measurement values within a single document. Alternatively, you could specify that installation instructions for Linux or MacOS be excluded and then output content using specific named values for a Windows OS, such as Windows 7, Windows 8.1, or Windows 10.
Cascade Attribute

In DITA 1.2, metadata values were always additive by default. For example, if you added metadata values to the audience for a map (platform="Windows MacOS") and then add a separate targeted topicref to that map containing audience="Linux", the latter value would automatically be added to the other metadata values for that map. When the map is published, all of the Windows, MacOS, and Linux topics are included. But what if that was not the intention and the writer wanted to specify that the topicref for audience="Linux" should not be included? The @cascade attribute in DITA 1.3 allows writers to disable this default behavior. If the previous Linux value were appended with cascade="nomerge", the default cascading behavior would be turned off. Content creators now have more flexibility when running processing routines based on metadata.

Content-Sensitive Help

DITA support for various forms of Context-Sensitive Help (CSH) systems has been enhanced, allowing content creators to set the attributes for a specific callback ID and its context. Positional information for a particular target display can be set using the new <ux-window> element within the topicmeta of a map or as an attribute within a topic. Writers can also establish the priority setting of a CSH instruction that could, for example, ensure that the @ux-window attribute value specified within a topic overrides a more general setting for that same CSH system referenced at the map level. Help practitioners can now encode callback IDS and target window definitions for multiple target Help environments within the same set of DITA source files.

deliveryTarget (Goodbye print, Hello deliveryTarget)

The @print attribute has always had two practical limitations:

• Its binary "yes/no" nature was too limiting to address the increasingly diverse range of available output options
• The processor determined the filtering behavior. For a given setup, if PDF was excluded, XHTML was automatically included, which might not be what was intended.

In DITA 1.3 the @print attribute has been replaced with @deliveryTarget, allowing for any conceivable output type or device to be targeted. @deliveryTarget acts as a filtering attribute with all the expected controls, bells, and whistles that come with it. @deliveryTarget accepts one or more CDATA values, and information architects can use subjectScheme attributes to control @deliveryTarget values. More generally, it allows information architects to preprocess content targeted for specific output types. If you want to format your content specifically for HTML, HTML5, OpenOffice, Word, WebHelp, ePub3, Kindle, or more, you can now create specific output routines to do so.

Using @deliveryTarget does not remove the need to create specific transforms for each output type. Instead @deliveryTarget is designed to be a convenient way to choose from a variety of possible output options other than just @print.

Filtering Attributes Can Now Be Grouped
Writers have been limited to a single value for any given filtering attribute, so an individual item for audience might be set to audience="administrator" or audience="programmer" but not both at the same time. Because DITA 1.3 allows for grouped value filtering, a single item can now be set to multiple values (like audience="administrator programmer"). That same item can now be filtered against any of the values it contains. Similarly, values can also be placed into a parenthetical group whose individual values can be selected against. In the following example a specific OS could be filtered from the group: product="our_software (Windows Mac Linux)".

**Grouping Things Using div**

Writers can use the new block-level `<div>` element to create arbitrary groupings of content within a topic. It works in the same way as its HTML equivalent, defining an arbitrary block section. In HTML `<div>` is often used to group content which can then be formatted using CSS. In DITA its primarily purpose is to be used for organizing content that can then be referenced by a `@conref` or a `@conrefkey`. It can also be used by Information Architects as a convenient block-level base for specialization.

**Highlight Domain**

The highlighting elements are designed to render text in a particular style, such as bold `<b>`, italic `<i>`, superscript `<sup>`, subscript `<sub>`, teletype `<tt>`, or underline `<u>`. DITA 1.3 adds two new elements to this domain: `<line-through>` and `<overline>`. The `<line-through>` element indicates text that may no longer be relevant but needs to be displayed. Text marked up with `<overline>` is used in electrical engineering documentation to indicate active-low logic elements, and in a mathematical context to indicate that certain symbols belong together.

**Inline Elements for Describing Markup Language**

The new XML mention domain contains several inline elements created to describe XML markup:

- elements, using `<xmlelement>`
- attributes, using `<xmlatt>`
- namespace names, using `<xmlnsname>`
- processing instructions, using `<xmlpi>`
- numeric character entities, using `<numcharref>`
- parameter entities, using `<parameterentity>`
- text entities, using `<textentity>`.

For example, if you want to describe information about an XML element, you will no longer have to use entity characters for angle brackets (&lt; for <, and &gt; for >), but can instead wrap a pair of `<xmlelement>` elements around it and the output processor will apply the appropriate formatting.

**Key scope**
DITA 1.3 brings with it the ability to "scope" keys using the new @keyscope attribute, meaning that different key values can be "scoped" to specific locations within a map. If you have a topic that is being reused, you can specify different key values depending upon which sub-map that topic appears in. Key scopes help to maximize the reuse possibilities for keys in topics. It is now possible to use one key in a topic that reflects different values depending on the defined scope of the map containing it.

Scoped keys also allows for cross-deliverable linking. In DITA 1.3 you can define a map as being "peer" to another, so that the key values belonging to one map can be used by the other and vice-versa. Scoped keys opens the possibility of resolving keys from one map based on the key values set in a different peer map and generating cross-references to content contained within other maps.

With key scopes it is now easier to do the following:

- Combine multiple standalone publications with overlapping key names into a single, omnibus publication. Simply set the @keyscope attribute on the references to each map, preventing their keys from interfering with each other.
- Create a "mail merge" publication. Create a single map with key scope-defining wrappers around each instance of the template topic and the appropriate key definitions.
- Create multi-product documentation, where each section contains some common key names. Put key scopes around the parts of the map that pertain to different products.
- Use keys to create cross-references to content contained within a peer map.

Learning and Training Specialization

In DITA 1.2, writers were restricted to creating a single paragraph for question-and-answer exchanges in the Learning and Training specialization. Two new domains have been added to DITA 1.3—interactionBase2Domain and learning2Domain—that contain elements allowing for more robust multi-paragraph question, answer, and feedback interactions.

Writers can now also aggregate learning content, supporting files, and assessments into the new <learningObjectMap> element, and similarly, the <learningGroupMap> element can be used to collect together learning objects and group elements.

MathML and SVG Now Natively Supported

Before DITA 1.3, anyone using MathML (a specification for writing equations in XML) had to insert mathematical notations using the <foreign> element or by integrating the MathML standards into a customized version of their DITA DTD or Schema. This process was not easy and often involved repeated trial-and-error for those trying to make it work. MathML is now directly incorporated into DITA 1.3, so it can be used natively, using either the <mathml> element for containing MathML code directly within a topic, or <mathmlref> to reference the URI of a file containing MathML markup.

Similarly, inserting Scalable Vector Graphics (SVG) images—another, separate XML standard used for creating vector-based illustrations—has also been integrated with DITA 1.3 and can be used natively. Previously anyone wanting to use SVG images had to use the <image> element to point to them, and support for this graphic type was spotty with some output processors. Content creators wanting to insert SVG images into their topics can now use the <svg-container> element to hold native SVG code, or use <svgref> to reference the URI of SVG image file.

The expectation is that output processors previously incapable of handling MathML via the <foreign> element and SVG files via the <image> element will now be able to support these additional XML standards at output using the newly-introduced DITA 1.3 elements.
RELAX NG

The XML rules for previous versions of DITA were defined using Document-Type Definitions (DTDs) or W3C's XML Schema formats. With DITA 1.3, the rules are now written using RELAX NG (REgular LAnguage for XML Next Generation). Ideally, programs used for writing or processing DITA 1.3 content will be able to validate DITA topics based on this standard. RELAX NG offers greater flexibility, as its syntax is more robust and better suited to the task of defining the DITA constructs. DITA 1.3 will continue to be made available in DTD and Schema formats, but the RELAX NG version is the normative or ultimate version from which the others are derived.

Release Management

The new release management domain in DITA 1.3 provides the tools necessary for producing accurate release note content. Summary information on significant product additions or changes to a product can now be added to the prolog of a topic or within the topicmeta of a map using the release management elements as they are being written. Technical writers will no longer have to rely on memory alone to document the major changes to a release. The date and time information embedded in the release management content allows for scripts to programmatically select only those comments that are pertinent to a given release, while still keeping information about all successive significant changes over the development of a product. Additional notations that specify bug-tracking systems, identify change request details, and provide other release information allow organizations to publish the sources of significant changes to external systems.

Note that while the release management elements are available, as of the time of writing processing tools still need to be developed to output the release management content into a release note publication. The expectation is that tools will be developed and made available by the time DITA 1.3 is released or soon thereafter.

Same Topic References

In DITA 1.3, it is now possible to link to an ID located elsewhere within the same topic by using a period in the URI after the hash tag. If a writer wanted to link to a figure contained elsewhere in the topic, the link looks like this: `<xref="#./figure1">see figure 1</xref>`.

Sorting element

The new `<sort-as>` element is designed to help writers sort content where the base method for sorting is not obvious, such as for sorting words written in Chinese/Japanese/Korean characters where there is no inherent alphabetical order. So a writer can add the phonetic sound of a word and have the processor sort things like glossary entries, titles, or an index based on the spelled-out sound. For example the Chinese word for "hello" could be sorted in a glossary by its sound ("ni hao") as: `<glossterm><sort-as value="ni hao">你好</glossterm>`.

Table element—Turning Things Around

This is a Non-Standards Track Work Product and is not subject to the patent provisions of the OASIS IPR Policy.
Two attributes have been added to DITA 1.3 to aid writers who need more control over the print formatting of tables. 
<table> can now take the attribute @orient that instructs processors to print tables in landscape mode. Individual cell entries can take the new @rotate attribute that can turn content 90 degrees counterclockwise.

**Task for Troubleshooting**

The new <tasktroubleshooting> element has been added to the task topic type to assist readers who may encounter a common problem when attempting a task. This optional element appears at the conclusion of a task, along with <results>, <example>, and <postreq>. <tasktroubleshooting> must appear after a <result> and before an <example> and/or <postreq>.

Task topics can also take the optional troubleshooting element <steptroubleshooting>. <steptroubleshooting> can be used after <cmd> or <stepsresult> at the end of a particular <step> for situations when a user is likely to encounter a problem when performing that portion of a task.

**Troubleshooting Note Type**

The <note> element now has the new type attribute @trouble. This new note type informs the reader of a potential issue and how to resolve it. For example: <note type="troubleshooting">Use an ice-scraper to remove any ice and snow that has accumulated on the windshield.</note>

**Troubleshooting Topic**

DITA 1.3 adds a new topic type: troubleshooting. Its main structural elements (excluding elements common to other topic types) consist of the following:

- <condition> – describing a symptom that a user may encounter
- <troubleSolution> – one or more of these, describing a possible resolution to the issue
- <cause> – an optional follow-up for <condition> or <troubleSolution> describing a possible reason for the problem
- <remedy> – an optional, step-wise description of how to fix the problem, which can also include the optional <responsibleParty> element indicating who is expected to follow the remedy’s steps

The new troubleshooting topic is designed to do provide better structure for content based on the long-established troubleshooting pattern, instead of trying to shoe-horn it into another topic type. It also enables writers to focus on addressing and solving specific problems a user may encounter, and provides an opportunity for labeling troubleshooting content as such, making it easier for users to find. It also provides an ideal launching point for those wanting to specialize other flavors of troubleshooting topics.

**Other Updates to Individual DITA 1.3 Elements and Attributes**

- The <cite> element can be added to content within a <title>, allowing for citation references within a heading.
• The `<data>` element can now hold more comprehensive information, providing the option of adding citations, links, or short quotations to its content. `<data>` and the `<data-about>` elements can also be used as child elements of any list type, such as ordered or unordered lists.
• For many elements, the attribute values `@format` and `@scope`—commonly used in links—use the value "ditamap" so that links now point by default to the map rather than to the topic.
• Use of the `<draft-comment>` element has been expanded and now can be used in more places.
• The inline `<text>` element can now be used just about anywhere, and provides a flexible alternative to using `<ph>` when writers want to mark up phrases for reuse.
• Writers can now add `<ph>`, `<sub>`, and `<sup>` inline to individual `<indexterms>`.
• Keyrefs can now be added to the `<object>` and `<param>` elements.
• The `<prop>` and `<revprop>` elements can now take the `@style` attribute, so text selected in a DITAVAL file can be set to bold, italics, or any other style property.
• Titles can now take the `@rev` (revision) attribute.
• You can now add an `<xref>` link within a `<shortdesc>`.