Live question and answer session

Tuesday, 27 April
12-1:00 PM EDT

Contours of DITA 2.0

Kristen James Eberlein
Owner, Eberlein Consulting LLC
Chair, OASIS DITA Technical Committee

CONVEX VIRTUAL EXPERIENCE
APRIL 26-28, 2021

Agenda

1. Who designed DITA 2.0
2. DITA 2.0 as a backwards incompatible release
3. What’s in DITA 2.0
4. Getting ready for DITA 2.0

Who designed DITA 2.0?

The DITA Technical Committee

16 voting members who meet weekly

Wide variety of professional roles:

- Information developers
- Information architects
- Content engineers
- Managers
- Professors
- Consultants
- Application developers

We represent the community!

Why a backwards-incompatible release?

Why is DITA 2.0 backwards incompatible?

A backwards-incompatible release lets us:

- Reduce technical debt
- Get rid of unnecessary baggage
- Fix design mistakes
- Clean house!

Eek! You’ll break all my stuff!

- Yes, it’s likely that there will be changes that you’ll need to make to move to DITA 2.0.
- The DITA TC will provide migration documentation.
- Good opportunity to reassess your content, think about what you want to bring forward.

So, what’s in DITA 2.0?

Overview: Contours of DITA 2.0

- Architectural changes
  - Improvements to key-based variable text
  - Redesign of chunking
  - Relaxation of specialization rules
- Improvements to:
  - Book map
  - DITAVAL
  - <example>
  - Glossary elements
  - Hazard statements
  - Indexing
  - Simple table
  - <steps>
  - Troubleshooting topic

- New elements
  - <include>
  - Multimedia elements
- New domains:
  - Alternate titles
  - Emphasis
  - Hardware
- Behind-the-scenes changes
- Removal of:
  - Deprecated items
  - Attributes, elements, and domains
  - Support for XSD, certain specializations

Improvements to how variable text is defined

Use case

As a content author, I want a simple way to define variable text.

As an application developer, I want rules for resolving variable text to be simpler.

Return to overview

Implementation

The following changes:

- Add a new element: <keytext>
- Establish new rules for resolving variable text defined using key references

Champion
Kristen James Eberlein,
Eberlein Consulting LLC

Redesign of chunking

Use case

As a DITA map author, I want chunking to be easier to use.

As an application developer, I want chunking to be easier to implement.

Implementation

The following changes:

- Remove all previously-defined tokens for @chunk
- Replace with two new tokens: “combine” and “split”

Champion
Robert Anderson,
Oracle

Return to overview

Relaxation of specialization rules

Use case

As a DITA practitioner, I want to:

- Add a new specialized attributes ONLY to certain elements
- Add a new specialized element ONLY to a certain element

Return to overview

Implementation

The following changes:

- Enable attribute and element domain specializations to be more targeted
- Distinguish between document-type configuration and element configuration

Champion

Chris Nitchie, Individual member

Implementation work by Kristen James Eberlein

Improvements to book map

Use case

As a map author, I want to:

- Apply a DITAVAL to the entire bookmap
- Specify resource-only objects (key definition, subject scheme maps, etc) in an intuitive location
- Specify that a list of updates should be auto-generated

Return to overview

Implementation

The following changes:

- Modification to content models of <bookmap> and <booklists>
- New element in map group domain: <mapresources>

Champion
Kristen James Eberlein,
Eberlein Consulting LLC

Significant contributions by
Eric Sirois, IXIASOFT

Enhancements to flagging using DITAVAL

Use case

As a content author, I want a CSS class added to HTML output:

- When the DITA source is tagged with a filtering/flagging attribute
- A flagging action is set in the DITAVAL

Champion
Robert Anderson,
Oracle

Implementation

The following changes:

- Add @outputclass to the <prop> or <revprop> elements in DITAVAL
- Specify that that the expected behavior is that @outputclass token gets passed through in the HTML output as a @class value.

Return to overview

Make <example> available in more places

Use case

As a content author, I want to semantically tag content as an example where ever it occurs.

Implementation

We made the following changes:

- Made <example> available in more places
- Now available within sections, between paragraphs, in definition lists, figures, and more

Champion
Scott Hudson,
ServiceNow

Return to overview

Addition of <sub> and <sup> to glossary elements

Use case

As a content author, I want to be able to use subscripts and superscripts when I create glossary entries.

Implementation

The following changes:

- Allow <ph> within the content models of the relevant glossary elements
- Enables the use of the highlight domain

Champion
Scott Hudson, Service Now

Return to overview

Improvements to hazard statements

Use case

As a content author, I want to:

- Use ordered and unordered lists when specifying how to avoid a hazard
- Avoid @outputclass

As an information architect or manager, I want to ensure that our hazard statements meet industry requirements.

Return to overview

Implementation

Redesign of the hazard statement domain, including:

- Alignment with ANSI X 535.6
- Enable use of multiple hazard symbols
- Improvement to the specification

Champion
Kristen James Eberlein,
Eberlein Consulting LLC

Improvements to indexing

Use case

As a content author, I want:

- To stop inadvertently inserting <index-base> elements
- More reliable, consistent indexing support from processors
- Less complexity, more simplicity!

Implementation

The following changes:

- Removal of the indexing domain
- Addition of <index-see> and <index-see-also> to the base
- Removal of <index-base> and <index-sort-as>
- Improved specification content

Champion

Kristen James Eberlein,
Eberlein Consulting LLC

Return to overview

Enhancements to simple table

Use cases

As a content author, I want to:

- Span rows and columns in a simple table
- Add a title to a simple table

As a DITA practitioner, I want a useful base from which to specialize table types.

And the DITA TC wants to enable Lightweight DITA!

Return to overview

Implementation

The following changes:

- Add title to <simpletable>
- Enable column- and row spanning in simple tables

Champion

Carlos Evia, Virginia Tech

Improvements to <steps>

Use case

As a content author, I want to:

- Easily reuse <steps> and <step> elements
- Have more than two levels of steps
- Have unordered steps as a 2nd (or 3rd or 4th) level of steps

Return to overview

Implementation

We made the following changes:

- Enable <steps> to nest
- Remove <substeps> and <substep>

Champion
Robert Anderson,
Oracle

Diagnostic information in troubleshooting topic

Use case

As a content author, I want to add diagnostic information to troubleshooting topics in a semantic place.

Implementation

Three new elements:

- <diagnostics>
- <diagnostics-general>
- <diagnostics-steps>

Champion
Dawn Stevens,
Comtech Services, Inc.

Return to overview

New <include> element

Use cases

As a content author, I want to better handle the inclusion of code or text in my content.

As a DITA practitioner, I want a more appropriate specialization basis for transclusion elements.

Return to overview

Implementation

The following changes:

- Addition of <include> element
- Changing the specialization basis of <coderef>, <svgref>, and <mathmlref> to <include>

Champion
Chris Nitchie,
Individual member

New multimedia elements

Use case

As a content author, I want to easily add audio and video to my DITA topics

Implementation

The following changes:

- <audio> and <video> elements
- Three sub-elements for controls

Champion
Kristen James Eberlein,
Eberlein Consulting LLC

Built on work completed by Chris Nitchie, Individual member

Return to overview

New alternate titles domain

Use cases

As a DITA practitioner, I want to specialize elements for titles: book sub-titles, window titles, etc.

As a content or map author, I want to stop being confused about when a navigation title will be rendered.

Champion
Chris Nitchie,
Individual member

Implementation

The following changes:

- Make <titlealts> available in maps
- New <titlealt> element available within <titlealts>
- Use <titlealt> as new specialization base for <searchtitle>, <navtitle>, and <linktitle>
- Remove @locktitle

Return to overview

New emphasis domain

Use case

As a DITA practitioner, I want a more semantic alternative to the highlighting domain.

Return to overview

Implementation

The following changes:

- Addition of an emphasis domain
- By default, added to all technical-content topic types

Champion

Keith Schengili-Roberts,
Precision Content

New hardware domain

Use case

As a content author, I want to:

- Have markup for hardware
- Stop using @outputclass to distinguish between different types of controls

Return to overview

Implementation

The following changes:

- Addition of a hardware domain with two elements: <hwcontrol> and <partno>
- Integrated into all technical content topics

Champion
Zoe Lawson,
Individual member

“Behind-the-scenes” changes

Return to overview

- Made @outputclass a universal attribute
- (Re)created @audience, @platform, @product, and @otherprops as specializations of @props
- Resolved inconsistent @class attribute values for <shortdesc>, <linktext>, and <searchtitle>
- Changed specialization basis for some elements, including <imagemap>
- Removed @xtrf and @xtrc attributes
- Split syntax and programming domain
- Removed @domains attribute; replaced with @specializations attribute

Champion

Robert Anderson, Oracle

Champion

Eliot Kimber, Individual member

Champion

Kris Eberlein, Eberlein Consulting LLC

Remove all deprecated items

Use cases

As the DITA TC, we want to:

- Reduce our technical debt
- Remove cruft from the standard
- Start with a fresh(er) slate
- Generally, clean house!

Return to overview

Implementation

Remove anything characterized as:

- Deprecated
- Reserved for future use
- Added by mistake and retained to ensure backwards compatibility

Champion
Kristen James Eberlein,
Eberlein Consulting LLC

Remove certain attributes, elements, and domains

- Remove the @copy-to and @lockmeta attributes
- Remove delayed conref domain
- Remove <topicset> and <topicsetref> elements
- Remove “fastpath” token from note/@type

Return to overview

Champion
Alan Houser, Individual member

Champion
Eliot Kimber, Individual member

Champion
Bill Burns, Healthwise

Champion
Keith Schengili-Roberts, Precision Content

Remove support for XSD, certain specializations

- Will not ship XSD for DITA 2.0

- Specializations removed from DITA 2.0
  - Learning and training
  - Machinery task
  - Task prerequisites domain

  - These specializations will be updated for DITA 2.0 and located in OASIS Open GitHub repositories

Return to overview

Getting ready for DITA 2.0

Getting ready for DITA 2.0: The big picture

There are lots of moving pieces in a DITA implementation; a backwards-incompatible DITA release potentially effects all of them:

1. **Current DITA source**
2. Information model
3. Document-type shells, specializations, constraints
4. Stylesheets
5. Tool stack

Assessing your DITA source

- Start with looking for (the most common) deprecated items
    - @alt attribute
    - @navtitle attribute
    - @title on <map>
    - @print attribute
- For a complete list, see the DITA 2.0 stage two proposal, #36 “Remove deprecated items.”
- **Tip:** Use Oxygen XML Editor’s DITA map metric report.

Assessing your DITA source

- Look for (non-deprecated) elements and attributes removed in DITA 2.0
  - <index-base>
  - <index-sort-as>
  - <substeps>
  - <substep>
  - <topicset> and <topicsetref>
  - @copy-to
  - @lock-title
  - @xtrf and @xtrc
  - Current tokens for @chunk
- Look for elements used in processing that are changing
  - Do you use <keyword> to define variable text referenced by keys?
- Look for whether your content uses elements or domains that have been redesigned, such as hazard statement

Assessing source: My advice

1. Get rid of deprecated items NOW.

2. Put guard rails in place to prevent content authors from reintroducing them

3. If an element is being removed in DITA 2.0, and a replacement element currently exists – Change your source to use the replacement element.

4. Consider appointing a DITA 2.0 captain, to build expertise and knowledge.

5. Start considering whether you'll want to take advantage of DITA 2.0 enhancements, for example:
    - Ability to use <example> in more places
    - If you use troubleshooting topics, new <diagnostic> element

Attend the live conversation!

Tuesday, 27 April 2021
12-1:00 PM EDT
Everest

And if you are looking at a PDF of this presentation, here is my contact information:

Kristen James Eberlein
kriseberlein on Skype
kris@eberleinconsulting.com
+1 919 622-1501

Branded color palette