
Stage two (revised): #29 <bookmap> update

Revise book map design to remediate problems but avoid breaking backwards compatibility.

Date and version information

This proposal includes the following information:

Date that this feature proposal was completed

14 October 2019

Champion of the proposal

Kristen James Eberlein, Eberlein Consulting LLC

Links to any previous versions of the proposal

There were several earlier versions of this stage two proposal:

- [26 February 2018](#)
- [04 February 2019](#)
- [19 February 2019](#) (approved by TC)

Links to minutes where this proposal was discussed at stage 1 and moved to stage 2

[Minutes, 23 May 2017](#)

Reviewers for stage two proposal

This revised proposal was reviewed by the following voting members:

Robert Anderson, IBM
Nancy Harrison, Individual member
Eliot Kimber, Individual member
Dawn Stevens, Comtech Services, Inc

In addition, the following TC member provided review feedback:

Leigh White, IXIASOFT

Links to e-mail discussion that resulted in new versions of the proposal

The following e-mails resulted in this stage two proposal, approved 19 February 2019, being revised and resubmitted to the DITA TC:

Date	Author	Title	Date discussed at TC meeting
18 September 2019	Kristen James Eberlein	Question about issue 29 "Bookmap update"	01 October 2019
19 September 2019	Kristen James Eberlein	Concerns about issue #29 "Bookmap updates"	01 October 2019
01 October 2019	Kristen James Eberlein	Bookmap updates and the glossary reference domains	08 October 2019
01 October 2019	Kristen James Eberlein	Info from minutes about adding glossary reference domain to bookmap	08 October 2019

Date	Author	Title	Date discussed at TC meeting
02 October 2019	Kristen James Eberlein	How best to handle <mapresources>	08 October 2019
07 October 2019	Kristen James Eberlein	Integrate glossary reference domain into bookmap?	08 October 2019
07 October 2019	Kristen James Eberlein	Content model and attributes for <mapresources>	08 October 2019

On 08 October 2019, the DITA TC decided that the issue of integrating the glossary reference domain into bookmap should be considered as part of a separate proposal that focuses on glossary design.

Link to the GitHub issue

[#29 Modify bookmap design](#)

Original requirement or use case

The requirements and use cases for this proposal were part of a larger discussion about bookmap, its deficits, and the development of a new publication map that began in early 2017. For excerpts from all TC meetings minutes that pertained to this discussion, see this [PDF](#).

E-mails that specifically pertained to remediating the bookmap design include the following:

Date	Author	Requirement or use case
21 February 2017	Eric Sirois, IXIASOFT	Limitations for keyref and ditavalref in bookmap
21 February 2017	Eric Sirois, IXIASOFT	Allow keyref in more natural position in bookmap
22 May 2017	Chris Nitchie, Oberon Technologies	Allow <ditavalref> before <frontmatter>
05 April 2018	Stefan Eike on dita-comment list	Allow <amendments> in <booklists>
25 April 2018	Eliot Kimber, Individual member	Bookmap does not include glossary reference domain

Use cases

This proposal will enable map authors to:

Apply a DITAVAL to the entire bookmap

The DITA design for bookmap and the DITAVALref domain makes it impossible for map authors to specify a DITAVAL that applies to the entire bookmap. In order for a <ditavalref> element to apply to the bookmap as a whole, the <ditavalref> element needs to be present at the root of the bookmap; this is not permitted in the current design of bookmap.

Specify resource-only objects in an intuitive location

The design for bookmap, developed for DITA 1.1, does not include an intuitive location for map authors to specify resource-only objects such as:

- Key definitions
- Subject scheme maps
- Topics that contains information used for the rendering of PDF cover pages

Adding a wrapper element with `@processing-role` set to "resource-only" will add a clear location for bookmap authors to add resource-only objects. To enable map authors to specify that the resources apply to the entire bookmap, the new element will be included in the content model for `<bookmap>`.

Because such an element will be useful in `<map>`, it will be defined in the map group domain. This also will make the new element available in bookmap specializations of `<topicref>` such as `<chapter>`, `<part>`, and `<appendix>`.

Specify that a list of updates should be auto-generated

DITA 1.3 introduced the release management domain, which enables authors to add change information to topics and maps. Some implementations developed plug-ins to harvest this information and generate a list of updates, but were stymied by the lack of a clear way to indicate in a bookmap that a list of updates should be generated.

Adding `<amendments>` to `<booklists>`, which is available in both front and back matter will address this issue. (In addition, this change addresses the fact that `<amendments>` is currently allowed only in back matter and many people want it in front matter also.)

New terminology

None

Proposed solution

- Modify the content model of `<bookmap>` to allow zero or more `<ditavalref>` elements after `<bookmeta>`
- Add a `<mapresources>` element to the map group domain. The `<mapresources>` element will:
 - Be specialized from `<topicref>`
 - Have a content model of:
 - Zero or one `<topicmeta>` elements
 - Zero or more `%data.elements.incl;` elements
 - Zero or more `<topicref>` elements
 - Have the following attributes: `@href`, `@keys`, `@keyref`, `@keyscope`, `@processing-role`, `@type`, `@cascade`, `@scope`, `@format`, `@linking`, `@toc`, `@search`, and `%univ-atts;` attributes
 - The attributes should have the same values as `<keydef>`, with the following exception: the `@keys` attribute is not required.
- Modify the content model of `<bookmap>` to allow zero or more `<mapresources>` elements after `<ditavalref>`
- Modify the content model of `<booklists>` to allow zero or more `<amendments>`

Note This solution adds hard dependencies on two domains to the bookmap module: DITAVALref and map group. This is something that the TC has not previously done in document-type shells and will need to be clearly documented and explained to DITA practitioners.

Benefits

This proposal addresses the following questions:

Who will benefit from this feature?

Information architects and map authors.

What is the expected benefit?

- An easier-to-use bookmap that enables information architects and map authors to:
 - Apply a DITAVAL to the entire bookmap
 - Locate key definitions, subject scheme, and other resource-only objects (such as cover page information) in an intuitive location
 - Indicate to processors where an auto-generated change list should be generated
- A more robust map group domain, that includes a convenience element for containing resource-only object

How many people probably will make use of this feature?

Many or most bookmap authors; many map users

How much of a positive impact is expected for the users who will make use of the feature?

Significant

Technical requirements

This proposal involves the following changes:

Adding an element

This proposal will add a `<mapresources>` element to the map group domain. This element will:

- Serve as a wrapper element to hold key definitions, subject schemes, and other resource-only objects
- Have the `@processing-role` attribute set to "resource-only"
- Be specialized from `<topicref>` and be available where ever `<topicref>` is permitted

The `<mapresources>` element cannot include text, and it does not define new elements or attributes that can contain translatable text.

Refactoring an element

The content models of two elements defined in the bookmap module will be modified:

`<bookmap>`

- Zero or more `<ditavalref>` elements allowed after `<bookmeta>`
- Zero or more `<mapresources>` elements allowed after `<ditavalref>`

`<booklists>`

Zero or more `<amendments>` elements allowed.

Processing impact

None

Overall usability

Improved usability for current and future users

Backwards compatibility

This proposal addresses the following questions:

Was this change previously announced in an earlier version of DITA?

No

Removing a document type that was shipped in DITA 1.3?

No

Removing a domain that was shipped in DITA 1.3?

No

Removing a domain from a document type shell that was shipped in DITA 1.3?

No

Removing or renaming an element that was shipped in DITA 1.3?

No

Removing or renaming an attribute that was shipped in DITA 1.3?

No

Changing the meaning of an element or attribute in a way that would disallow existing usage?

No

Changing a content model by removing something that was previously allowed, or by requiring something that was not?

No

Changing specialization ancestry?

No

Removing or replacing a processing feature that was defined in DITA 1.3?

No

Are element or attribute groups being renamed or shuffled?

No

Migration plan

Implementations will need to pay attention to their document-type shell for book map. The DITA 2.0 specialization for bookmap requires that the DITAVAlref and map group domains be integrated.

Costs

This proposal has a (time and effort) impact on the following groups:

Maintainers of the grammar files

The following files will need to be modified:

- DTDS:
 - bookmap.dtd
 - bookmap.mod
 - mapGroup.mod
 - mapGroup.ent
- RNG:
 - bookmap.rng
 - bookmapMod.rng
 - mapGroupDomain.rng

Note

The following comment must be added to the bookmap document-type shells.

```
<!-- Do not remove references to the map group or DITAVAlref -->  
<!-- domains. The bookmap specialization requires these domains. -->
```

Editors of the DITA specification

Note All numeric references here are to the DITA 2.0 specification, [working draft 11](#).

The changes to the DITA specification will involve a medium amount of effort:

- A new `<mapresources>` topic.
- A new topic for the "Examples of constraints" section. This topic will illustrate how a DITA practitioner would modify a bookmap document-type shell to remove the DITavalref domain.
- Adding `<mapresources>` to the "DITA elements, A to Z" topic.
- Updates to the "Example" section in the `<bookmap>` topic to show:
 - Usage of `<ditavalref>`
 - Usage of `<mapresources>`
 - Usage of `<amendments>` within `<booklists>`
- Updates to the `<amendments>` topic to explain processing expectations for an `<amendments>` element that does not reference a resource.
- Updates to the short description in the `<booklists>` topic.
- 3.4.3 "DITA map elements" (Not clear if this topic will remain in the DITA 2.0 specification).

In addition, the following topics will be examined to determine whether they should be reworked to include the new `<mapresources>` element:

- 6.4.10 "Examples of keys" and its child topics
- 6.4.11 "Examples of key scopes" and its child topics
- `<map>` topic

This proposal does not require changes to either the specification information architecture or terminology.

Vendors of tools

Tools that render DITA maps will need to modify their CSS or XSLT to specify how the `<mapresources>` element is rendered.

Processors will need to implement support for generating/rendering update lists based on the presence of the `<amendments>` element within `<booklists>`.

DITA community-at-large

The changes to bookmap and the map group domain should not add to a perception of DITA complexity. The changes should be simple and intuitive for end users to understand.

Producing migration instructions or tools

Not applicable

Examples

This section provides examples of the new markup enabled by this proposal.

Figure 1: Applying a DITaval to an entire bookmap

The following code sample shows how a bookmap can specify a `<ditavalref>` element that applies to the entire bookmap:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE bookmap PUBLIC "-//OASIS//DTD DITA 2.0 BookMap//EN" "bookmap.dtd">
<bookmap>
  <booktitle>
    <mainbooktitle>Test publications</mainbooktitle>
```

```

    </booktitle>
    <ditavalref href="test.ditaval"/>
    ...
</bookmap>

```

Figure 2: Specifying resource-only objects in an intuitive location in a book map

The following code sample illustrate how the new `<mapresources>` element can group references to key definitions, subject schemes, and other resources in a bookmap:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE bookmap PUBLIC "-//OASIS//DTD DITA 2.0 BookMap//EN" "bookmap.dtd">
<bookmap>
  <booktitle>
    <mainbooktitle>Test bookmap</mainbooktitle>
  </booktitle>
  <mapresources>
    <mapref href="key-definitions.ditamap"/>
    <mapref href="subject-scheme.ditamap" type="subjectscheme"/>
    <topicref href="cover-page.dita outputclass="cover-page"/>
  </mapresources>
  ...
</bookmap>

```

Note that this example illustrates that `<mapresources>` can be used to make topics available for resource-only processing. In this scenario, the company uses a processor that uses content contained in the `cover-page.dita` file to generate a PDF cover page.

Figure 3: Specifying resource-only objects in a map

The following code sample shows a map that contains information for a specific model of a controller. This map is referenced in an omnibus publication that contains information for an entire family of controllers.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE map PUBLIC "-//OASIS//DTD DITA 2.0 Base Map//EN" "basemap.dtd">
<map keyscope="model-XNP09">
  <title>Model XNP09</title>
  <mapresources>
    <keydef keys="model-illustration" href="model-XNP09.png" format="png"/>
    <keydef keys="remove-cover" href="remove-cover-XNP09.png" format="png"/>
  </mapresources>
  <topicref href="model-overview.dita"/>
  <topicref href="installing.dita"/>
  <topicref href="uninstalling.dita"/>
  ...
</map>

```

Figure 4: Indicating that a processor should generate a list of changes

The following code sample specifies that a processor should auto-generate a list of changes and render it at the location of the `<amendments>` element:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE bookmap PUBLIC "-//OASIS//DTD DITA 2.0 BookMap//EN" "bookmap.dtd">
<bookmap>
  <booktitle>
    <mainbooktitle>Test bookmap</mainbooktitle>
  </booktitle>
  <frontmatter>
    <booklists>
      <amendments/>
    </booklists>
  </frontmatter>

```

```
...  
</bookmap>
```

If the `<amendments>` element specifies a DITA resource, then processors will use that as source for the change list.