DITA 2.0 proposed feature #670 Impose map role

Define a new attribute `@impose-role` that can be used on topicref elements. When specified on a topicref element with a value of "impose", and the topicref element is a reference to a map, this indicates that the role of the topicref overrides the role of the referenced map or topicref.

Tracking information

In the following table, insert cross references to e-mails to the TC list or meeting minutes:

- Be sure to use the public URL as the value of the `@href` attribute
- For link text for the cross references, use "E-mail, date month year", "GitHub issue", or "Minutes, date month year"

Event	Date	Links
Initial suggestion	01 March 2022	Minutes, 01 March 2022
Stage 1 proposal accepted	15 March 2022	Minutes, 15 March 2022 https://github.com/oasis-tcs/dita/issues/670
Stage 2 proposal submitted to TC for early feedback (not applicable to all proposals)	24 June 2022	https://lists.oasis-open.org/archives/dita/202206/msg00057.html
Stage 2 proposal submitted to reviewers	28 June 2022	Carsten Brennecke, SAP Gershon Joseph, Precision Content
Stage 2 proposal submitted to TC	<xref to e-mail sent to list>	
Stage 2 proposal discussed by TC	<date>	<xref to meeting minutes where discussed>
Stage 2 proposal approved by TC	<date>	<xref to meeting minutes where approved>

Original requirement or use case

The idea for this proposal emerged at the DITA TC meeting on 01 March 2022, as a result of a discussion of open items from DITA 2.0 review K: "Metadata and cascading metadata".

From DITA 1.0, the specification has contained a section on cascading of "roles" from one map to another when a `<topicref>` is used as a reference to another map. The specific and most common example is the `<chapter>` element, which can be used to reference other maps or branches in other maps, effectively imposing its own role of "chapter" on the referenced branches. This is described here: Cascading of roles from map to map.

The biggest problem with this ~~is that behavior~~ is that the specification says that it happens with specialized elements "unless otherwise instructed" – but there is no way to instruct a processor when this behavior is desired. It also says any non-default behavior "should be clearly specified", but we do not provide a way to specify it. From the spec:

> Unless otherwise instructed, a specialized `<topicref>` element that references a map supplies a role for the referenced content. This means that, in effect, the `@class` attribute of the referencing element cascades to top-level topicref elements in the referenced map. In situations where this should not happen - such as all elements from the mapgroup domain - the non-default behavior should be clearly specified.

This feature request for DITA 2.0 is meant to fill that gap, allowing the vocabulary to specify this behavior, so that processors can identify how they are supposed to handle the imposition of roles.

Use cases

When used, the new @impose-role attribute will most often be set up as a default attribute value in a vocabulary module, and will not be set or seen by content authors.

When a topic reference does not refer to a map or map branch, the attribute will be ignored. For all other cases, the attribute will serve to indicate whether the map reference pushes its role onto the target when resolving that reference, as described in the spec today for the <chapter> element.

As a specific example, the <chapter> element will be defined with this attribute set to a default value of "impose". This means that when the <chapter> element refers to a map or map branch, the referenced content should itself be treated as a chapter - effectively imposing the role of - map/topicref bookmap/chapter onto the referenced content.

In the following example, because of the new @impose-role attribute set for <chapter> in the bookmap grammar module, a processor would know:

- To treat each top-level topic reference in othermap.ditamap as a chapter on its own, whtether or not they are already <chapter> elements.
- To treat the referenced branch thirdmap.ditamap#reusebranch as a chapter on its own, whether or not it is already a <chapter> element.
- To pull in references from summary.ditmap using normal processing, without imposing any special role.

```
<bookmap>
    <booktitle>
        <mainbooktitle>Sample book map</mainbooktitle>
    </booktitle>
    <chapter href="othermap.ditamap" format="ditamap"/>
    <chapter href="thirdmap.ditamap#reusebranch" format="ditamap"/>
    <chapter href="closing-chapter.dita">
        <topicref href="summary.ditamap" format="ditamap"/>
    </chapter>
</bookmap>
```

New terminology

n/a

Proposed solution

We will define a new attribute on <topicref> and each of its specializations. The attribute will be named @impose-role and will have the following values:

- "keeptarget": when the reference is to a map or map branch, this reference does not impose any role on the target. This is the processing default and equivalent to not specifying the attribute.
- "impose": when the reference is to a map or map branch, this reference imposes its own role on the target.
- "-dita-use-conref-target" the usual conref token

The attribute will be specified on each <topicref> in our vocabulary with one of the two values. It will be set to "keeptarget" for the base <topicref> element and for all specializations in the map group domain. It will also be set to this value for each <topicref> in the subject scheme map. It will be set to "impose" for each <topicref> in the bookmap module.

The attribute does not cascade.

The attribute does not resolve as part of a key reference. For example, the `<keydef>` attribute will have a default value of "keeptarget", which should not override the default that is already set up on a `<chapter>` or other `<topicref>` that happens to refer to the key.

Benefits

Address the following questions:

Who will benefit from this feature?

Information architects have a clear way to declare a behavior that is currently described, but impossible to define.

Processors will know what to do and will not require customizations or one-off code as needed today for bookmaps.

What is the expected benefit?

More consistent processing, clearer specification behavior, and ability to use an already-defined DITA feature that is not really available outside of bookmaps today.

How many people probably will make use of this feature?

Many, although most will do so unknowingly because it will largely be based on default attributes in grammar files.

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

Minor.

Technical requirements

Provide a detailed description of how the solution will work. Be sure to include the following details:

Adding new elements or attributes

Adding an attribute

- Name of the attribute: `@impose-role`
- Elements for which the attributes will apply: `<topicref>` and all specializations of `<topicref>`
- Processing expectations that are associated with the new attribute: when set to "~~overriderole~~", it declares the processing that is already described in 2.2.4.6.3 Cascading of roles from map to map
- Does the attribute contain translatable text? No

Processing impact

- How will the feature work?

  When set to "~~overriderole~~", the new `@impose-role` attribute declares that a processor should implement the processing that is already described in 2.2.4.6.3 Cascading of roles from map to map
- Will the feature have an impact on other processing features? For example, will the proposed feature have an impact on key resolution? No
- Will the feature have to be evaluated before or after any existing features? No
- What edge cases need to be considered? No

Overall usability

Most DITA users will not notice this attribute, as it is set up as a default within grammar files.

Information architects that ~~was~~ this behavior will now be able to declare this behavior in a clear and logical way when defining their own maps.

Backwards compatibility

No impact on backwards compatibility.

Migration plan

N/A

Costs

Outline the impact (time and effort) of the feature on the following groups.

Maintainers of the grammar files
    Small cost, declaring the attribute.

Editors of the DITA specification

- How many new topics will be required? This will require reworking the existing topic: 2.2.4.6.3 Cascading of roles from map to map, it will also require extensive reworking the example topic on cascading of roles (new topic already present in DITA 2.0 draft).
- How many existing topics will need to be edited?

  - The "common attributes" topic will need to declare this attribute
  - Each `<topicref>` will need to refer to it, with many using the default but bookmap using the override value

- Will the feature require substantial changes to the information architecture of the DITA specification? No
- If there is new terminology, is it likely to conflict with any usage of those terms in the existing specification? No

Vendors of tools
    Tools that follow the spec behavior for `<chapter>` today likely have overrides that already set up this sort of processing; those can be simplified to rely on the attribute and use a common behavior.

DITA community-at-large

- Will this feature add to the perception that DITA is becoming too complex? No
- Will it be simple for end users to understand? Probably not, but they will generally not encounter it directly
- If the feature breaks backwards compatibility, how many documents are likely to be affected, and what is the cost of migration? N/A

Producing migration instructions or tools
    N/A

Examples

Provide examples of the proposed feature. Include an example for each of the use cases. Be sure to include edge cases, if known.

In the following example, because of the new `@impose-role` attribute set for `<chapter>` in the bookmap grammar module, a processor would know:

- To treat each top-level topic reference in `othermap.ditamap` as a chapter on its own, whtether or not they are already `<chapter>` elements.
- To treat the referenced branch `thirdmap.ditamap#reusebranch` as a chapter on its own, whether or not it is already a `<chapter>` element.
- To pull in references from `summary.ditmap` using normal processing, without imposing any special role.

```
<bookmap>
    <booktitle>
        <mainbooktitle>Sample book map</mainbooktitle>
```

```
        </booktitle>
    <chapter href="othermap.ditamap" format="ditamap"/>
    <chapter href="thirdmap.ditamap#reusebranch" format="ditamap"/>
    <chapter href="closing-chapter.dita">
        <topicref href="summary.ditamap" format="ditamap"/>
    </chapter>
</bookmap>
```