
Stage three: #647 Replace classification domain with a new attribute

Replace classification domain with a new attribute: @subjectrefs

Champion

Kristen James Eberlein, Eberlein Consulting LLC

Tracking information: Stage two

Event	Date	Links
Initial suggestion	E-mail, 09 December 2021	Minutes, 04 January 2022
Stage 1 proposal accepted	11 January 2022	Minutes, 11 January 2022 GitHub issue
Stage 2 proposal submitted to TC for early feedback (not applicable to all proposals)	Not applicable	
Stage 2 proposal submitted to reviewers	E-mail, 07 March 2022	Carsten Brennecke, SAP Gershon Joseph, Precision Content I've also requested feedback from the following members of the larger DITA community: <ul style="list-style-type: none">• Peyton Bentley, Kaplan• Joe Gelb, Zoomin Software• Pam Noreault, Ellucian• Shane Taylor, Cengage Group
Stage 2 proposal submitted to TC	E-mail, 17 March 2022 E-mail, 29 March 2022	
Stage 2 proposal discussed by TC	22 March 2022 29 March 2022	Minutes, 22 March 2022 Minutes, 29 March 2022
Stage 2 proposal approved by TC	05 April 2022	Minutes, 05 April 2022
Stage 2 proposal reopened	24 May 2022 The proposal was reopened to allow @subjectrefs on <map>.	
Revised stage 2 proposal submitted to reviewer		Not applicable
Revised stage 2 proposal submitted to TC	E-mail, 26 May 2022 E-mail, 29 May 2022	
Revised stage 2 proposal approved	31 May 2022	Minutes, 31 May 2022

Tracking information: Stage three

Event	Date	Links or notes
Stage 3 proposal submitted to reviewers	E-mail, 04 June 2022	Robert Anderson, Oracle Carsten Brennecke, SAP Gershon Joseph, Precision Content Eliot Kimber, ServiceNow This proposal was reviewed in Content Fusion, June 2022. In addition, reviewers held two phone calls to resolve differences.
Stage 3 proposal submitted to TC	E-mail, 13 July 2022	
Stage 3 proposal discussed	<Date>	<xref to meeting minutes where discussed>
Stage 3 proposal approved	<Date>	<xref to meeting minutes where discussed>

Approved technical requirements: Stage two

The following is the content that was approved at stage two:

Remove the classification domain and the classification map. Add a new attribute, @subjectrefs, to the <topicref> element and specializations of <topicref>, as well as <map> and specializations of <map>. This includes both elements in the base and the technical content edition.

Note The @subjectrefs attribute has no meaning if it is specified on a key definition that does not reference a resource.

The @subjectrefs attribute will have the following characteristics:

- It will cascade.
- It will take multiple values, separated by white space. The values are one or more keys, defined in a subject scheme map.

While the DITA TC does not want to introduce any processing expectations for @subjectrefs in the DITA 2.0 time frame, it is possible that we will introduce them in the future.

Technical requirements: Stage three

Remove the classification domain and the classification map. Add a new attribute, @subjectrefs, to the <topicref> element and specializations of <topicref>, as well as <map> and specializations of <map>. This includes both elements in the base and the technical content edition.

The @subjectrefs attribute **has** the following characteristics:

- It **cascades**.
- It **takes** multiple values, separated by white space. The values are one or more keys, **each** defined in a subject scheme map.

While the DITA TC does not want to introduce any processing expectations for @subjectrefs in the DITA 2.0 time frame, it is possible that we will introduce them in the future.

Dependencies or interrelated proposals

None.

Removed grammar files

The following grammar files need to be removed. In addition, the catalog files that currently reference the following files will need to be modified.

Base

- doctypes/dtd/subjectScheme/classifyDomain.ent
- doctypes/dtd/subjectScheme/classifyDomain.mod
- doctypes/rng/subjectScheme/classifyDomain.rng

Technical content

- doctypes/dtd/classificationMap directory
- doctypes/rng/classificationMap directory

The classification domain and the document-type shell for a base classification map will be loaded into the GitHub repository for specializations that are removed from the DITA standard.

Modified grammar files

This proposal will require modifications to the following files:

Base edition

- dtd/base/map.mod
- dtd/base/mapGroup.mod
- rng/base/map.rng
- rng/base/mapGroupDomain.rng

Technical content edition

- dtd/technicalContent/glossrefDomain.mod
- rng/technicalContent/glossrefDomain.mod

No changes are required to the bookmap grammar files, since the relevant attribute definitions reference the topicref attribute entities that are defined in the base grammar files.

In the content below, the following conventions are used:

- Bold is used to indicate code to be added, for example, **addition**.
- Line-through and red text is used to indicate code to be removed, for example, ~~removal~~.
- Ellipses (...) indicate where code is snipped for brevity.

Figure 1: Changes to map.mod

```
<!ENTITY % topicref-atts
    "...
    subjectrefs
    CDATA
    #IMPLIED"
>
...
<!ENTITY % topicref-atts-without-format
    "...
    subjectrefs
    CDATA
```

```
> #IMPLIED"
```

Figure 2: Changes to mapGroup.mod

```
<!ENTITY % keydef.attributes
    "...
    subjectrefs
        CDATA
        #IMPLIED
    %univ-atts;"
>
```

Figure 3: Changes to mapMod.rng

```
<div>
  <a:documentation>COMMON ATTRIBUTE SETS</a:documentation>

  <define name="topicref-atts">
    ...
    <optional>
      <attribute name="subjectrefs"/>
    </optional>
  </define>
  ...
  <define name="topicref-atts-without-format">
    ...
    <optional>
      <attribute name="subjectrefs"/>
    </optional>
  </define>
```

Figure 4: Changes to mapGroupDomain.rng

```
<div>
  <a:documentation>Key Definition</a:documentation>
  ...
  <define name="keydef.attributes">
    ...
    <optional>
      <attribute name="subjectrefs"/>
    </optional>
    <ref name="univ-atts"/>
  </define>
```

Figure 5: Changes to glossrefDomain.mod

```
<!ENTITY % glossref.attributes
    "...
    subjectrefs
        CDATA
        #IMPLIED
    %univ-atts;"
>
```

Figure 6: Changes to glossrefDomain.rng

```
<div>
  <a:documentation>ELEMENT TYPE DECLARATIONS</a:documentation>

  <div>
    <a:documentation> LONG NAME: Glossary Reference </a:documentation>
    ...
    <define name="glossref.attributes">
      ...
    </optional>
```

```

<optional>
  <attribute name="subjectrefs"/>
</optional>
<ref name="univ-atts"/>
</define>

```

Modified terminology

Not applicable

Modified specification documentation

The following topics need to be removed or modified:

Topics and maps to be removed

- archSpec/base/classification-maps.dita
- langRef/classification-domain-elements.ditamap
- langRef/containers/classify-d.dita
- langRef/base/subjectCell.dita
- langRef/base/subjectref.dita
- langRef/base/topicapply.dita
- langRef/base/topicCell.dita
- langRef/base/topicsubject.dita
- langRef/base/topicSubjectHeader.dita
- langRef/base/topicSubjectRow.dita
- langRef/base/topicSubjectTable.dita

Architectural topics and maps to be modified

File	Modification
cascading-in-a-ditamap.dita	Add @subjectrefs to the list of attributes that cascade.
cascading-of-attributes-from-map-to-map.dita	Add @subjectrefs to the list of attributes that cascade from map to map.
subjectSchema.dita	Remove the following sentence: "In conjunction with the classification domain, subject definitions can be used for retrieval and traversal of the content at run time when used with information viewing applications that provide such functionality."
subject-scheme-maps.ditamap	Remove reference to classification-maps.dita. Add references to new topics: <ul style="list-style-type: none"> • The subjectrefs attribute (6) • Example: A subject scheme map used to define taxonomic subjects (7)
example-cascade-map-to-map-attributes.dita	Modify the content as shown in Example: How attributes cascade from one map to another (7) .

Element-reference topics and maps to be modified

The following attribute definition for @subjectrefs will need to be added to the appropriate attribute topic:

Specifies one or more **keys** that are **each** defined **by a subject definition** in a subject scheme map. Multiple values are separated by white space.

Base edition

The following topics will need to include the `@subjectrefs` attribute:

- `<keydef>`
- `<map>`
- `<mapref>`
- `<topichead>`
- `<topicgroup>`
- `<topicref>`

This will happen automatically by updating the content in the "Attributes" topic for the "common map attributes."

Technical content edition

The following topics will need to include the `@subjectrefs` attribute:

- `<glossref>`
- All bookmap elements that are defined with the `topicref-atts` entity

This might happen automatically depending on what attribute "grouping" the `@subjectrefs` attribute is part of.

Topics to be added

The following topics are to be added to the "DITA map processing" chapter, in the "Subject scheme maps and their usage" section:

- [The subjectrefs attribute \(6\)](#)
- [Example: A subject scheme map used to define taxonomic subjects \(7\)](#)

The following content, which has not been edited, is to be considered for inclusion in the keys topic cluster of the "DITA processing" chapter: [Additional examples with key scopes \(8\)](#)

Migration plans for backwards incompatibilities

Implementations that use the classification domain and classification map will need to do one of the following, if they want to move to DITA 2.0:

- Download the classification domain from the GitHub repository and integrate it into the relevant document-type shells. If the implementation currently uses the OOB OASIS-provided document-type shell for the classification map, they will need to move to using a custom document-type shell.
- Replace map markup that uses the classification domain with the new `@subjectrefs` attribute, as well as modifying any processing that is based on the classification domain.

The `@subjectrefs` attribute

The `@subjectrefs` attribute specifies one or more keys that are defined by a subject definition in a subject scheme map. Multiple values are separated by white space.

The `@subjectrefs` attribute cascades. **When specified on a topic reference, the `@subjectrefs` attribute associates the referenced resource with subjects that are defined in subject scheme maps.**

The DITA 2.0 specification does not indicate processing expectations for the @subjectrefs attribute. The DITA Technical Committee expects to specify such expectations in the future.

Example: A subject scheme map used to define taxonomic subjects

A subject scheme map can be used to define taxonomic subjects. Once defined, the subjects can be referenced by specifying a @subjectrefs attribute on a <topicref> element.

The following subject scheme map defines a set of subjects that are used to classify content:

```
<subjectScheme>
  <subjectdef keys="content-types">
    <subjectdef keys="conceptual-material"/>
    <subjectdef keys="reference"/>
    <subjectdef keys="tutorial"/>
  </subjectdef>
  <subjectdef keys="operating-systems">
    <subjectdef keys="linux"/>
    <subjectdef keys="macosx"/>
    <subjectdef keys="windows"/>
  </subjectdef>
  <subjectdef keys="user-tasks">
    <subjectdef keys="administering"/>
    <subjectdef keys="developing"/>
    <subjectdef keys="installing"/>
    <subjectdef keys="troubleshooting"/>
  </subjectdef>
</subjectScheme>
```

The keys assigned to the subject definitions can be referenced by specifying the @subjectrefs attribute on topic references in a navigation map:

```
<map>
<title>User assistance for the Acme Widget</title>
<!-- ... -->
<topicref keyref="install-overview" subjectrefs="installing">
  <topicref keyref="install-linux"/>
  <topicref keyref="install-macosx"/>
  <topicref keyref="install-windows"/>
  <topicref keyref="install-troubleshooting" subjectrefs="troubleshooting"/>
</topicref>
<!-- ... -->
</map>
```

Because the @subjectrefs attribute cascades, the effective value of the above markup is the same as the following markup:

```
<map>
<title>User assistance for the Acme Widget</title>
<!-- ... -->
<topicref keyref="install-overview" subjectrefs="installing">
  <topicref keyref="install-linux" subjectrefs="installing"/>
  <topicref keyref="install-macosx" subjectrefs="installing"/>
  <topicref keyref="install-windows" subjectrefs="installing"/>
  <topicref keyref="install-troubleshooting" subjectrefs="installing troubleshooting"/>
</topicref>
<!-- ... -->
</map>
```

Example: How attributes cascade from one map to another

In this scenario, attributes in one map cascade to a nested map.

Assume the following references in `test.ditamap`:

```
<map>
  <topicref href="a.ditamap" format="ditamap" toc="no"/>
  <mapref href="b.ditamap" audience="developer"/>
  <mapref href="c.ditamap#branch2" platform="myPlatform"/>
  <mapref href="d.ditamap" subjectrefs="puzzles"/>
</map>
```

- The map `a.ditamap` is treated as if `toc="no"` is specified on the root `<map>` element. This means that the topics that are referenced by `a.ditamap` do not appear in the navigation generated by `test.ditamap`, except for branches within the map that explicitly set `toc="yes"`.
- The map `b.ditamap` is treated as if `audience="developer"` is set on the root `<map>` element. If the `@audience` attribute is already set on the root `<map>` element within `b.ditamap`, the value "developer" is added to any existing values.
- The element with `id="branch2"` within the map `c.ditamap` is treated as if `platform="myPlatform"` is specified on that element. If the `@platform` attribute is already specified on the element with `id="branch"`, the value "myPlatform" is added to existing values.
- The map `d.ditamap` is treated as if `subjectrefs="puzzles"` is set on the root `<map>` element. If the `@subjectrefs` attribute is already set on the root `<map>` element within `d.ditamap`, the value "puzzles" is added to any existing values.

Additional examples with key scopes

Comment by Kristen J Eberlein on 13 July 2022

The following is content that Eliot Kimber suggested be included in the DITA 2.0 specification. It has not been edited. It will be edited as part of the DITA 2.0 editorial process, and this content is slated for inclusion in the chapter on "DITA processing," not the chapter on "DITA map processing".

A subject scheme map may be included in a map as either a normal sub map or as a peer root map and associated with a key scope on the map reference.

Figure 7: A subject scheme map included as a sub map with a key scope:

```
<map>
  <title>User assistance for the Acme Widget</title>
  <!-- ... -->
  <mapref keyscope="subjects" href="subjectschemes/subject-scheme-01.ditamap"/>
  <topicref keyref="install-overview" subjectrefs="subjects.installing">
    <topicref keyref="install-linux"/>
    <topicref keyref="install-macosx"/>
    <topicref keyref="install-windows"/>
    <topicref keyref="install-troubleshooting" subjectrefs="subjects.troubleshooting"/>
  </topicref>
  <!-- ... -->
</map>
```

The keys defined in referenced subject scheme map are included in the referencing map's key space. References to the subject keys must be scope qualified (i.e., "subjects.installing"). Putting the subject scheme keys in a key scope ensures that the keys defined in the subject scheme do not conflict with any keys defined elsewhere in the referencing map (although the referencing map could still override any keys defined in the subject scheme map).

The subject scheme map can also be referenced as a peer-scope root map in a key scope:

```
<map>
  <title>User assistance for the Acme Widget</title>
  <!-- ... -->
  <mapref keyscope="subjects" scope="peer" href="subjectschemes/subject-scheme-01.ditamap"/>
  <topicref keyref="install-overview" subjectrefs="subjects.installing">
    <topicref keyref="install-linux"/>
    <topicref keyref="install-macosx"/>
    <topicref keyref="install-windows"/>
    <topicref keyref="install-troubleshooting" subjectrefs="subjects.troubleshooting"/>
  </topicref>
  <!-- ... -->
</map>
```

As a peer map, the keys defined in the subject scheme map are not included in the referencing map's key space but may be resolved using normal cross-deliverable key reference resolution. Processors that support cross-deliverable linking may also choose to resolve references from `@"subjectrefs"` to keys in peer subject scheme maps, for example to report keys specified on `@"subjectrefs"` that cannot be found in the peer subject scheme map.