

Review A 2024: Glossentry elements

Table of contents

1 Topic and map document types.....	3
1.1 Glossary entry.....	3
1.2 Glossary group.....	5
2 Glossary entry topics and terminology management.....	9
2.1 abbreviated-form elements and glossary entry topics.....	10
2.2 Rendering of <abbreviated-form> elements.....	10
3 Glossary entry elements.....	13
3.1 <glossAbbreviation>.....	13
3.2 <glossAcronym>.....	14
3.3 <glossAlt>.....	16
3.4 <glossAlternateFor>.....	18
3.5 <glossBody>.....	20
3.6 <glossdef>.....	21
3.7 <glossentry>.....	21
3.8 <glossPartOfSpeech>.....	23
3.9 <glossProperty>.....	25
3.10 <glossScopeNote>.....	26
3.11 <glossShortForm>.....	28
3.12 <glossStatus>.....	28
3.13 <glossSurfaceForm>.....	29
3.14 <glossSymbol>.....	31
3.15 <glossSynonym>.....	32
3.16 <glossterm>.....	33
3.17 <glossUsage>.....	34
A Aggregated RFC-2119 statements.....	37
B Attributes.....	38
B.1 Attribute groups.....	38
B.2 Universal attribute group.....	51
B.3 Common attributes.....	56
B.4 STUB CONTENT.....	68
Index.....	70

1 Topic and map document types

The Technical Content package contains various document types: concept, glossary entry, glossary group, reference, general task, strict task, and troubleshooting. The package also includes the **book map** document type.

1.1 Glossary entry

A glossary entry topic defines a single meaning

Comment by Eliot Kimber on 2024-01-28

Do we still want to say "single meaning here or should be it be "one or more meanings"?

[Kris Eberlein, 05 March 2024] Yes, we want to say a single meaning. If a term has multiple meanings (concepts), there should a separate glossentry topic for each meaning.

Disposition: Closed

of a term. It also can provide information such as part of speech

Comment by Amber Swope on 07 Feb 2024

If there are multiple definitions for a term in a single glossentry topic, then how do you associate usage information to a given definition?

For example, if definition 1 applies to product A and definition 2 applies to product B, then how do you identify the context and align glossScopeNote or glossUsage to a specific definition?

[Kris Eberlein, 05 March 2024] One definition, one glossentry topic.

Disposition: Closed

, acronyms, and acronym expansions.

Purpose

Glossary entry topics serve the following purposes:

- They ensure that a team of writers can use the same terminology.
- They make it possible to create glossaries that can be use to provide readers with definitions of terms and explanations about acronyms.

Comment by Leroy Steinbacher

Consider just "They can be used to create glossaries that provide readers with definitions of terms and explanations about acronyms.

WEK: I think I agree. While it's true that a glossary can serve as a terminology references for a writing team, that doesn't seem like a driving or primary purpose (there are lots of ways to do that, including terminology management tools like Acrolinx).

Nancy: Leroy, did you mean to remove the first list item, or just to edit the second one? I can't tell from your comment. Not every writing team has access to good terminology tools, so I'm not sure I'd remove the first item.

Leroy: I meant just edit the 2nd list item.

[Kris Eberlein, 05 March 2024] Correct the typo, leave the first list item as-is.

[Kris Eberlein, 24 March 2024]: Done

Disposition: Completed

Comment by Zoë Lawson

I'm not entirely sure this is an implementation feature that we don't want to talk about, but for me one of the selling points of going through all the trouble of making special glossary files is that I can also easily set up links to the glossary terms. Whether that gets rendered as a simple link or a happy pop up that automagically gives the definition of the term, that seems to be a huge value add. It's not just a formatting thing so I can generate something that looks like a glossary. Probably a user guide idea again.

[Kris Eberlein, 05 March 2024] I don't think we want to call out the specific functionality of using `<term>` + glossentry topics to generate links, since that is basic hyperlinking strategy. It would be a good item to call out in a user-guide type reference that helped people choose what DITA elements to use in their content models. But the spec is not that sort of reference.

Disposition: Closed

Content model

Each glossary entry topic contains the following structures:

- Term

Comment by Eliot Kimber on 2024-01-28

Would like to add the tag names in parens: "Term (`<glossterm>`)" and "Definition of term (`<glossdef>`)"

[Nancy Harrison] I agree with Eliot, and would add the same comment for glossary body `<glossbody>`

[Kris Eberlein, 05 March 2024] We need to be consistent with what we have done with the other topics in this section of the spec. I'm off to take a look.

The general task, reference, and strict task topics have the clearest and most consistent pattern. We will implement that across all of these "conceptual" topics in the technical content spec.

[Kris Eberlein, 24 March 2023]: Done

Disposition: Completed

- Definition of term
- Prolog
- Glossary body (which can contain acronyms and surface terms)
- Related links

Comment by Tony Self

Prolog is referenced in the glossgroup topic, so this topic should use a parallel construction. (I'm assuming glossentry/prolog is still in 2.0?)

[Kris Eberlein, 05 March 2024] We need to be consistent with what we have done with the other topics in this section of the spec. I'm off to take a look.

The general task, reference, and strict task topics have the clearest and most consistent pattern. We will implement that across all of these "conceptual" topics in the technical content spec.

Marking the disposition of this comment to "Closed, " as we will track the work under the draft comment above.

Disposition: Closed

Example

The following code sample contains a simple glossentry topic:

```
<glossentry id="ddl">
  <glossterm>data definition language</glossterm>
  <glossdef>A language used for defining database schemas</glossdef>
</glossentry>
```

Comment by Kristen J Eberlein on 07 October 2022

This should have a glossbody that specifies an acronym and a surface term. Also related links.

[Kris Eberlein, 05 March 2024] This is a draft comment that existed before the Content Fusion review. I do think we should have two examples in this topic: one for a simple topic and the second for a glossentry topic that is used in conjunction with <abbreviated-form>, and so uses <glossSurfaceForm> and <glossAcronym>.

[Robert Anderson, 14 March 2024] Agreed, we should have an example of both here.

[Kris Eberlein, 24 March 2024] Done

Disposition: Completed

1.2 Glossary group

Glossary group topics enable the authoring of glossary entries in a single topic document, rather than working with many individual glossary entry topic documents.

Comment by Leroy Steinbacher

Not sure about the use of "document" above. Would "file" be more appropriate (as used below)?

WEK: "document" is the more correct term as "file" is not necessarily how XML documents are stored. As XML, and by extension, DITA, are W3C standards, documents are "resources" (things addressed by URIs), and "file" is only one way to store them. So even though we all use "file" informally and know what it means, it's not the most correct term in the context of this specification.

[Kris Eberlein, 05 March 2024] Very much agree with Eliot. @Eliot, thanks for the clear explanation – You've made my work here in the review much easier.

Disposition: Rejected

Purpose

The glossary group topic type is primarily an authoring convenience where it is useful to author and manage multiple glossary entries in a single file

Comment by Eliot Kimber on 2024-01-28

Per the above, "file" should be "XML document" or "resource".

[Kris Eberlein, 05 March 2024] Agree.

[Kris Eberlein, 24 March 2024] Done.

Disposition: Completed

Note that it is possible to create separate topicrefs, such as `<glossref>`, for individual glossary entry topics within a glossary group topic

Comment by Eliot Kimber on 2024-01-28

Change to "within the topic reference to a glossary group topic". As written, it implies the topicrefs can go in the topic itself, which of course they cannot.

[Kris Eberlein, 05 March 2024] This paragraph is confusing and poorly worded. I suspect that the original authoring intent was to specify that one could address an individual glossentry topic located within a `<glossgroup>` topic by `<topicref>`.

Perhaps reword as "Note that it is possible to address individual glossary entry topics within a glossary group topic by topic reference."

I think introducing `<glossref>` here overly complicates the matter.

[Robert Anderson, 14 March 2024] Agree with Kris's comments. I think the intent here was to highlight that you could have a topic reference to a specific entry (that is, a `topicref` to a specialized topic) – something that is true whether or not we say it, and seemed worth highlighting. I'm not sure we need the paragraph given that it doesn't actually create a new behavior, and seems to be adding confusion.

[Kris Eberlein, 24 March 2024] Removed the entire example.

Disposition: Completed

, for example:

```
<map>
...
  <topicref keys="glossary" href="glossary/glossary-group.dita">
    <glossref keys="term-01" href="glossary/glossary-group.dita#term-01"/>
    <glossref keys="term-02" href="glossary/glossary-group.dita#term-02"/>
    ...
    <glossref keys="term-99" href="glossary/glossary-group.dita#term-99"/>
  </topicref>
</map>
```

Comment by Zoë Lawson

I'm finding this example confusing because I have no idea what this does or why I would do it. I can create the separate topicrefs (or glossrefs), but why? (Also the sudden introduction of glossref was more confusing than helpful. And it's okay to have the key name match the id? They don't have to be unique?) Keys are awesome and wonderful, but they sometimes make my brain hurt.

Ah. Worth mentioning "Use this when defining keys for glossary entries that work with abbreviated-form"?

[Kris Eberlein, 05 March 2024] I suggest that we remove the example. I don't think it really adds to this conceptual overview of <glossgroup>.

[Robert Anderson, 14 March 2024] Agree, along with comments above - I think this example is adding confusion.

[Kris Eberlein, 24 March 2024] Removed the example; marking this comment as closed.

Disposition: Closed

Content model

A glossary group topic has a required title, an optional prolog, and zero or more <glossentry> or <glossgroup> elements.

Example

The following code sample shows a glossary group topic with multiple nested glossary groups, one for each English letter group.

Comment by Zoë Lawson

Should we rephrase? "which could be used to have a nested glossary group for each English letter group" or something like that? Stating the example has a glossgroup for each letter when we only show two doesn't seem that quite right.)

[Kris Eberlein, 05 March 2024]

I think we need a simpler example here, not one with nested glossary groups! It's a clever example, but I think we need to simply illustrate the more common uses of <glossgroup>.

[Robert Anderson, 14 March 2024] Should refer to TC to get suggestion for better example.

Disposition: Referred

```
<glossgroup id="glossgroup" xml:lang="en-US">
  <title>Glossary</title>
  <glossgroup id="glossgroup-a">
    <title>A</title>
    <glossentry id="apple-fruit">
      <glossterm>apple</glossterm>
      <glossdef>A round, edible fruit produced by an apple tree (Malus domestica).</glossdef>
    </glossentry>
    <glossentry id="apple-corp">
      <glossterm>Apple Inc.</glossterm>
      <glossdef>An American multinational technology company headquartered in Cupertino,
California.</glossdef>
      <glossBody>
        <glossAlt>
          <glossAbbreviation>APPL</glossAbbreviation>
        </glossAlt>
      </glossBody>
    </glossentry>
  </glossgroup>
</glossgroup>
```

```

        </glossAlt>
      </glossBody>
    </glossentry>
  </glossgroup>
  ... (groups B to Y here ) ...
  <glossgroup id="glossgroup-z">
    <title>Z</title>
    <glossentry id="ziziphus-fruit">
      <glossterm>ziziphus</glossterm>
      <glossdef>The edible drupe of ziziphus shrubs (Ziziphus jujuba).</glossdef>
      <glossBody>
        <glossAlt>
          <glossSynonym>jujube</glossSynonym>
        </glossAlt>
      </glossBody>
    </glossentry>
  </glossgroup>
</glossgroup>

```

2 Glossary entry topics and terminology management

Comment by Tony Self

We don't seem to have covered the relationship between `<term>` and `<glossentry>`. For example, `<term keyref="p_cefazolin">` is linked to a glossentry topic by `<keydef href="quicklinks/glossary_topics/g_p_cefazolin.dita" keys="p_cefazolin"/>`. (Presented in HTML output as a popup or lightbox to glossary definition.) If this is covered under `<term>`, then a link back to that explanation might be useful.

[Scott Hudson] I agree with Tony, that we should provide an explanation of the relationship or a link to such explanation.

[Kris Eberlein, 05 March] @Tony, I think what you bring up here is straight-forward hyperlinking. I'm not sure if we need to address it here. Of course, I'm also not sure what exactly we do need to add here!

I added this section to the draft spec because we badly needed to move the content about rendering `<abbreviated-form>` out of the element-reference topic for `<abbreviated-form>`. I'm ready to rewrite that topic, based on a conversation at today's DITA TC call. Hopefully when that topic is reworked, we'll have a better sense of what needs to be covered in this section.

[Robert Anderson, 14 March 2024] We definitely don't want the specification instructing that there are associated pop-ups for these. This is an implementation of the spec that would be optional. I've seen this behavior disabled by some; it's similar to using keys to pull in text for a keyword, which can optionally turn into a link and/or have the same pop up behavior.

[Kris Eberlein, 24 March 2024] Marking this comment as "Closed".

Disposition: Closed

Comment by nancylph

Are we planning on adding any general introductory language about our perspective on how to use glossary topics for terminology mgmt? e.g., in a shortdesc for this topic? Is there anything in the arch. spec now?

[Kris Eberlein, 05 March 2024] This is 100% new content for DITA 2.0, and obviously we did not get the content into the topic before this review.

[Robert Anderson, 14 March 2024] Recent TC discussion has removed most of the older elements here, making this far from a terminology management solution - I think that reduces the need to give our perspective on that, just because we've stripped out most markup that could have been associated with that.

[Kris Eberlein, 24 March 2024] Changed the title of the topic to "Usage of glossary entry topics". The only content is the short description: "Glossary entry topics are designed for several purposes. In addition to providing the basis for glossaries, they also can be used in conjunction with the abbreviated-form domain to provide different renderings of terms on first and later usage."

Disposition: Completed

Comment by Amber Swope

Are there points that we might leverage from the glossary white paper from years past?

[Tony Self] This is the text from the whitepaper:

The linking of a term occurring inline in a content topic to its relevant glossary definition is a matter of associating the term (semantically identified in a term element) with the hypertext reference (href) or the key reference (keyref) of the glossary topic. For example, the markup `<term keyref="eoy">EOY</term>` establishes the association between the term "EOY" and the glossary topic with a keys attribute of eoy. The related map entry may be `<glossref keys="eoy" href="eoy_nz.dita"/>`

[Kris Eberlein, 24 March 2024] Based on discussion in comment earlier in the topic – and the disposition of those comments, marking this comment as "Closed".

Disposition: Closed

2.1 abbreviated-form elements and glossary entry topics

Comment by Christina Rothwell on January 23, 2024

This section is at the same level as the next section, but this section contains no content. Are we planning to add content to this topic?

[Kris Eberlein, 05 March 2024] Yes, we will add content here.

[Kris Eberlein, 24 March 2024] Removed this topic.

Disposition: Closed

2.2 Rendering of <abbreviated-form> elements

There are specific rules that specify how processors should render `<abbreviated-form>` elements.

Comment by Robert Johnson

How about "DITA includes rules that specify..." "There are" is a weak construction, and "specific rules specify" is redundant.

[Scott Hudson] Agree with Robert on this shortdesc, but guidelines/suggested behavior instead of rules

[Kris Eberlein, 05 March 2024] I'm going to rework the entire topic. We are going to talk about "design intent" rather than rules.

[Kris Eberlein, 24 March 2024] I've reworked the topic. The short description now states "The `<abbreviated-form>` element is designed to work with glossary entry topics in specific ways." The normative rules are introduced with the following sentence: "The design intent is that processors use the following logic:".

Disposition: Completed

Comment by Kristen J Eberlein on 15 August 2023

I have a LOT of problems with these normative SHOULD statements:

- How can we have normative rules when the whole idea of introductory and non-introductory context is so entirely subjective?
- If we compare these to the normative rules for key text resolution that are specified in the base spec, we are in trouble. The base rules suggest that the title of the glossary entry topic is what should be used ... See [Processing key references to generate text or link text](#).

WEK: I'm not sure the variable key resolution rules apply as any reference to a glossary entry is, fundamentally, a cross reference to a topic, not a reference to a keydef with no referenced resource. The way that topics are reflected in cross references is entirely a rendition-time concern, so anything we say here can only be a suggestion.

[Robert Anderson, 14 March 2024] Not sure I agree with "anything we say here can only be a suggestion". How abbreviated content renders is core to the element itself, almost as much as making bold content **bold** is core to that element. We are certainly allowed to make normative recommendations here. That said:

- I would agree we cannot say MUST. It is rendering. Applications are free to break from recommendations.
- I don't even see any way to define "first usage" with normative rules. Which effectively makes all of this rather lax.
- Even if we go with a SHOULD about displaying the full form in first usage – applications can break that if they want to.

[Kris Eberlein, 24 March 2024] Removed the draft comment. FYI – It's note a comment made during this review; the value of the @time attribute indicates that it dates back to August 2023! Marking this comment as closed.

Disposition: Closed

Note that the definition of "introductory context" will vary based on the processor and output format.

Comment by Robert Johnson

Recommend deleting "Note that".

[Kris Eberlein, 05 March 2024] Agree

[Kris Eberlein, 24 March 2024] I've reworked the topic, and it no longer contains the phrase "Note that".

Disposition: Completed

Comment by Zoe Lawson

What the heck is "introductory context"? Ignore the element if it's at the end of a file? I find this statement confusing. I think it needs to be defined. Hrm. I found a bit more information in the abbreviated-form topic, but I'm not positive that's enough of a definition.

[Kris Eberlein, 05 March 2024] I agree, and the reworked version of this topic will be better.

[Kris Eberlein, 24 March 2024] The reworked topic no longer contains the phrase "introductory context".

Disposition: Completed

Comment by Zoë Lawson

I think this is the third time I'm stating this, but if abbreviated-form only works with glossAcronym, we should really state that.

[Robert Anderson, 14 March 2024] I think it's probably helpful to make this explicit – if you want to have different rendering, you need to make use of acronym / surface form.

[Kris Eberlein, 24 March 2024] I've made this explicit in the reworked topic.

Disposition: Completed

1. If the referenced topic is not a `<glossentry>` topic or a specialization of `<glossentry>`, the title of the topic *SHOULD* be rendered.
2. If the referenced topic is a `<glossentry>` topic or a specialization of `<glossentry>` and the `<abbreviated-form>` element is located in an introductory context:
 - (If the referenced topic contains a non-empty `<glossSurfaceForm>` element) Processors *SHOULD* render the contents of the `<glossSurfaceForm>` element
 - (If the referenced topic does not contain a non-empty `<glossSurfaceForm>` element) Processors *SHOULD* render the contents of the `<glossterm>` element
3. If the referenced topic is a `<glossentry>` topic or a specialization of `<glossentry>` and the `<abbreviated-form>` element is located in a non-introductory context:
 - (If the referenced topic contains a non-empty `<glossAcronym>` element) Processors *SHOULD* render the abbreviated form of the term by displaying the contents of the `<glossAcronym>` element.
 - (If the reference topic does not contain a non-empty `<glossAcronym>` element) Processors *SHOULD* render the contents of the `<glossterm>` element

Comment by Kristen J Eberlein on 15 August 2023

The rendering looks dreadful in PDF format ...

ZLL: Would it be worth making a flowchart/decision tree?

[Kris Eberlein, 24 March 2024] No, I'm just venting. The issue is really with how we display the normative rules in the PDFs. Marking this as "Closed."

Disposition: Closed

3 Glossary entry elements

The glossary entry specialization contains markup that supports terminological information.

Comment by Eliot Kimber on 2024-01-28

"lexicographical" may be the better term than "terminological":

<https://www.merriam-webster.com/dictionary/lexicography>

Likewise, "lexical" below should probably be "lexicographical"

[Kris Eberlein, 05 March 2024] This short description needs to be revisited AFTER we remove the element that we recently agreed to get rid of.

[Kris Eberlein, 24 March 2024] Done.

Disposition: Completed

In addition to recording information for glossaries, it can be used for other purposes such as inclusion in a terminology database, extraction for pre-translation, reuse in controlled authoring applications, and recording information that may be later retrieved for other lexical resources.

3.1 <glossAbbreviation>

The <glossAbbreviation> element provides an abbreviation of the term specified in the topic's <glossterm> element. Abbreviations are typically formed by omitting some letters from a longer form.

Comment by Zoë Lawson

Is that last sentence really necessary? (Also, some abbreviations may be weird due to translations, e.g. why it took me a long time to figure out Au for gold and Ag for silver.)

Since my brain groups abbreviations and acronyms together, is it worth highlighting that glossAbbreviation does not work with Abbreviated-form? (especially since the abbreviation doesn't work with abbreviated form? ow.)

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Specialization hierarchy

The <glossAbbreviation> element is specialized from <title>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [ID and conref attributes](#) (51), [localization attributes](#) (52), [@base](#) (52), [@class](#) (52), [@outputclass](#) (54), and [@rev](#) (55).

Example

The following code sample shows how the `<glossAbbreviation>` is used in a glossary entry topic:

```
<glossentry id="adjective">
  <glossterm>adjective</glossterm>
  <glossdef>A word that describes a noun or noun phrase</glossdef>
  <glossBody>
    <glossAlt>
      <glossAbbreviation>adj</glossAbbreviation>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.2 `<glossAcronym>`

The `<glossAcronym>` element defines an acronym for the term specified in the topic's `<glossterm>` element. An acronym typically is composed of the initial letters of the components of the full form or from the syllables of the full form to create a pronounceable combination of letters.

Comment by Robert Johnson

Should we also discuss initialisms?

I might say "acronym or initialism" just for completeness and to satisfy the tech writing nerds in our audience.

[Kris Eberlein, 05 March 2024] I think the short description might need to be reworked since we have removed `<glossAbbreviation>` and `<glossShortForm>`.

Here's the distinction that the CDC draws: "An abbreviation is a truncated word; an acronym is made up of parts of the phrase it stands for and is pronounced as a word (ELISA, AIDS, GABA); an initialism is an acronym that is pronounced as individual letters (DNA, RT-PCR). For the purposes of this section, "abbreviation" will refer to all of these."

I think that `<glossAcronym>` needs to be used for all three: abbreviation, acronym, and initialism.

[Robert Anderson, 14 March 2024] If initialisms are a subset of acronyms, then we should not over-complicate this by trying to add them in. I had not previously heard the word, and I think it is not necessary to explain that this applies to acronyms as well as a certain subset of acronyms.

[Kris Eberlein, 24 March 2024] I modified the short description to read as follows:

The `<glossAcronym>` element defines an acronym for the term that is specified in the `<glossterm>` element.

Disposition: Completed

Comment by Zoe Lawson

Do we need the last sentence? Should I ask why we differentiate between abbreviation and acronym?

[Kris Eberlein, 05 March 2024] Because the original design, introduced in DITA 1.2, made that clear distinction and believed it necessary to have both `<glossAbbreviation>` and `<glossAcronym>`?

Disposition: Closed

Usage information

This element can be used with the `<abbreviated-form>` element to display an expanded version of an acronym the first time that acronym appears in a set of text. See [<abbreviated-form>](#) for information on how the two elements interact.

Note An acronym in one language might not have an equivalent in another language. When acronyms are first displayed, some languages will display the expanded form first followed by the acronym in parenthesis, while other languages do the reverse. For some acronyms, a translation might need to render both the original and the translated version of the acronym. For all of these reasons, DITA allows an author or translator to control what is presented to a reader by using the `<glossSurfaceForm>` element, which will often accompany the `<glossAcronym>`. The [<abbreviated-form>](#) topic contains information on how the `<glossSurfaceForm>` and `<glossAcronym>` elements affect references to the term.

Comment by Eliot Kimber on 2024-01-28

Another note about localization implications. This seems like content that should be gathered in a separate "localizing glossaries" topic or maybe even a separate white paper.

[Kris Eberlein, 05 March 2024] I think we need to remove the this content from here. If we want to cover some of the localization challenges presented by `<abbreviated-form>` and `<glossSurfaceForm>` we should do that in the architectural topics about glossary and `<abbreviated-form>`

[Robert Anderson, 14 March 2024] Agreed, better in the architectural topic

[Kris Eberlein, 24 March 2024] Moved content into a new architectural topic.

Disposition: Completed

Specialization hierarchy

The `<glossAcronym>` element is specialized from `<title>`. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [ID and conref attributes](#) (51), [localization attributes](#) (52), [@base](#) (52), [@class](#) (52), [@outputclass](#) (54), and [@rev](#) (55).

Example

The following code sample shows how the `<glossAcronym>` element can be used:

Comment by Zoë Lawson

Would it be worth changing this example to use Doctors without Borders/Médecins Sans Frontières since it shows the translation acronym issue? or did I miss the point because that's more about Word (acronym) vs (acronym) Word?

[Robert Anderson, 14 March 2024] I don't think that's appropriate here. The language ref topic should have a simple working example. We will likely be addressing translation issues in the architectural spec component of this (tech-comm) spec, and if we have examples that **don't** work they should be part of that.

[Kris Eberlein, 24 March 2024] Marking this comment as "Rejected".

Disposition: Rejected

```
<glossentry id="united-nations">
  <glossterm>United Nations</glossterm>
  <glossdef>The United Nations, referred to informally as the UN, is an
    intergovernmental organization whose stated purposes are to maintain
    international peace and security, develop friendly relations among
    nations, achieve international cooperation, and serve as a center for
    harmonizing the actions of nations.</glossdef>
  <glossBody>
    <glossSurfaceForm>United Nations (UN)</glossSurfaceForm>
    <glossAlt>
      <glossAcronym>UN</glossAcronym>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.3 <glossAlt>

The `<glossAlt>` element contains information about a variant for the term specified in the topic's `<glossterm>` element. A variant might include an acronym, abbreviation, or shortened form of the term, as well as synonyms or other alternate terms.

Usage information

The variant should have the same meaning as the term in the `<glossterm>` element; the variant is simply another way to refer to the same term. There might be many ways to refer to a term; each variant is placed in its own `<glossAlt>` element. The `<glossUsage>` element can be used within `<glossAlt>` to indicate when use of the alternate term is appropriate.

Comment by Zoë Lawson

Dumb question, that's probably something that would be handled by the validator, but can I put text inside of `<glossAlt>`, or should content always be inside of other elements?

[Kris Eberlein, 05 March 2024] Content must be within elements that are children of `<glossAlt>`.

Disposition: Closed

Note A list of alternative terms is specific to the language, so the translation of a `<glossentry>` topic might result in empty elements within a `<glossAlt>` container.

Comment by Eliot Kimber on 2024-01-28

Not sure why it's necessary to have a localization comment here. We don't talk in general about the implications of localization on content elsewhere as far as I can remember and whether or not it results in empty elements or elements being removed is entirely a function of how the localization is performed.

[Kris Eberlein, 05 March 2024] We should remove this here, and devote some attention to localization issues in the architectural topics.

[Kris Eberlein, 24 March 2024] Done

Disposition: Completed

Specialization hierarchy

The <glossAlt> element is specialized from <section>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [universal attributes](#) (51).

Example

The following code samples shows how a glossary entry topic might provide alternative forms of the term.

```
<glossentry id="usa">
  <glossterm>United States of America</glossterm>
  <glossdef>A federal republic in the northern Western Hemisphere comprising 48 contiguous
    states, the District of Columbia, Alaska in North America, and Hawaii in the North
    Pacific, and in some contexts considered along with its five inhabited island
    territories (Puerto Rico, U.S. Virgin Islands, Guam, North Mariana Islands, American
    Samoa).</glossdef>
  <glossBody>
    <glossAlt>
      <glossAcronym>US</glossSynonym>
      <glossUsage>Used as an adjective.</glossUsage>
    </glossAlt>
    <glossAlt>
      <glossAcronym>USA</glossSynonym>
```

Comment by

Robert D Anderson

Need to clean up markup

[Kris Eberlein, 24 March 2024] I double-checked; the current markup is fine. The following markup validated in Oxygen:

```
<glossentry id="usa">
  <glossterm>United States of America</glossterm>
  <glossdef>A federal republic in the northern Western Hemisphere comprising 48
    contiguous states,
      the District of Columbia, Alaska in North America, and Hawaii in the North
    Pacific, and in
      some contexts considered along with its five inhabited island territories (Puerto
    Rico, U.S.
      Virgin Islands, Guam, North Mariana Islands, American Samoa).</glossdef>
  <glossBody>
    <glossAlt>
      <glossSynonym/>
      <glossUsage/>
    </glossAlt>
    <glossAlt>
      <glossSynonym/>
      <glossUsage/>
    </glossAlt>
  </glossBody>
</glossentry>
```

Disposition: Rejected

```
      <glossUsage>Used as a noun.</glossUsage>
    </glossAlt>
    <glossAlt>
      <glossSynonym>America</glossSynonym>
      <glossUsage>Do not use; the American continents include other countries.</glossUsage>
    </glossAlt>
    <glossAlt>
      <glossAbbreviation>the States</glossAbbreviation>
      <glossUsage>Informal. Typically used by English speakers outside of the country.</
glossUsage>
```

```
</glossAlt>
</glossBody>
</glossentry>
```

3.4 <glossAlternateFor>

The <glossAlternateFor> element indicates when a variant term has a relationship to another variant term in addition to the term specified in the topic's <glossterm> element.

Comment by dawnstevens

I am still struggling with this so we'll address it in the review. Couldn't <glossAlternativeFor> be used alone in a <glossAlt> element to indicate the <glossterm> of this topic is an alternative for another term defined in another topic? Why are we limiting the definition to indicate that a *variant* is an alternative for another term in addition to the current one?

WEK: I would expect term-to-term relationships to be marked up using related-links. But it could also make sense to allow just <glossAlternateFor> in <glossAlt> as an alternative to one of the <title> specializations. I agree that having this use limited specifically to variants seems overly constrained.

[Later] Having gotten to <glossSynonym>, I think that <glossSynonym> is the answer to Dawn's question: use <glossSynonym> to relate one glossary term as a whole to another and limit <glossAlternateFor> to just relating alternates to each other.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Usage information

Comment by Zoë Lawson

Why are we including usage information here, but not for glossAbbreviation or glossAcronym about it needing to be in <glossAlt>?

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

The <glossAlternateFor> element is available inside the <glossAlt> element, which is a container that provides a variant for the term specified in the topic's <glossterm> element. The <glossAlternateFor> element makes it clear that there is a relationship between the current topic's term, the variant term defined in the parent <glossAlt> element, and the term defined in the external glossary entry topic that it is linked to.

Specialization hierarchy

The <glossAlternateFor> element is specialized from <xref>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [link-relationship attributes](#) (56), [universal attributes](#) (51), and [@keyref](#) (62).

For this element, the @href or @keyref attribute references a term for which the current variant is an alternate (in addition to the <glossterm> of this <glossentry> topic). The reference could be to another <glossentry> topic or to a <glossAlt> element within the same <glossentry> topic (indicating that the current variant is an alternate for both the <glossterm> and the referenced alternate term).

Comment by Kristen J Eberlein on 06 August 2023

The above wording does not track for me, nor does it match the earlier example for <glossentry>, or the 2009 Kara Warburton white paper. You can read my interpretation in the explanation of the example below.

DS: I don't think the phrasing precludes your interpretation. In the example, fridge is an alternate for icebox and for refrigerator. But icebox could in fact be an alternate for refrigerator, so there could be a glossAlt establishing that, and then the glossAlternateFor from fridge could point to that glossAlt rather than another topic, and/or vice versa. I would think you would do that if you didn't want to have a separate topic for icebox.

DS: I did have to think through cases where all glossAlts might not be alternates for each other, though, to determine why you would do this. I decided if my term was airplane, plane might be an alt, as might 747 or 737, while plane might be an alternatefor 747 and 737, 747 would not be an alternate for 737. I don't know if I'm making sense, but this is how I've justified this element at all in my own head.

WEK: That seems pretty convoluted, something you'd want to capture in a subject scheme or relationship table--expressing those kinds of knowledge graph relationships seems pretty weak using xrefs and also something nobody ever do in practice--the authoring would be too hard the extraction of the knowledge too hard when you could set the same effect much more simply in other ways.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Example

The following code sample shows a glossary entry topic for the term "Refrigerator". In addition to providing a definition, the glossary entry topic defines a variant term ("fridge"). It also links to another glossary entry topic for a variant term ("icebox") which is defined in a separate glossary entry topic.

Comment by Eliot Kimber on 2024-01-28

Based on the discussion above, I think this example really demonstrates Dawn's point--"icebox" is clearly an alternative for "refrigerator" not just "fridge", given that "icebox" warrants its own glossary entry while "fridge" does not. There's an implicit data model that relates "refrigerator" to "icebox" (maybe "refrigerator" is a more general term of which "icebox" is a more specific instance in the data model).

It feels like the original designers of this markup were trying to provide a way to express what are essentially subject scheme relationships among terms and variants, but this is a very weak way to do it, and a mechanism that would likely not be generally actionable because there would be too much variance in how it was used, if at all, by different glossary authors or even different glossary entry writers.

If my concern as a lexicographer was to capture a detailed graph of relationships among terms and variants, this is not how I would do it. If I was a tech writer writing a glossary, I might use <glossAlternateFor> just to get readers to related terms, but I'd probably decide that was better done with <related-links>.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

```
<glossentry id="refrigerator">
  <glossterm>Refrigerator</glossterm>
  <glossdef>A refrigerator is a commercial and home appliance consisting
    of a thermally insulated compartment and a heat pump (mechanical,
    electronic or chemical) that transfers heat from its inside to its
    external environment so that its inside is cooled to a temperature
    below the room temperature.</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossAlt>
      <glossSynonym>fridge</glossSynonym>
      <glossUsage>Casual and colloquial usage only.</glossUsage>
      <glossAlternateFor keyref="glossentry-icebox"/>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.5 <glossBody>

The <glossBody> element contains information about the term specified in the topic's <glossterm> element, such as the part of speech and variants for the term.

Specialization hierarchy

The <glossBody> element is specialized from <conbody>; it is defined in the glossary entry module. The <conbody> element is specialized from <body>; it is defined in the concept module.

Attributes

The following attributes are available on this element: [universal attributes](#) (51).

Example

The following code sample shows how the <glossBody> element contains the part of speech and variant forms for the term:

```
<glossentry id="sport-drink">
  <glossterm>Sport drink</glossterm>
  <glossdef>A soft drink designed or marketed for consumption in
    conjunction with sporting activity or strenuous exercise, and which
    typically contains electrolytes such as sodium, potassium, and
    chloride, and a high percentage of sugar to restore energy.</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossAlt>
      <glossSynonym>energy drink</glossSynonym>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.6 <glossdef>

The <glossdef> element defines the meaning of the term specified in the topic's <glossterm> element.

Usage information

If a term has multiple concepts or meanings that need to be referenced separately, create a separate <glossentry> topic for each.

Comment by Eliot Kimber on 2024-01-28

Another instance of explicitly disallowing multiple definitions within a single <glossentry> topic.

[Kris Eberlein, 24 March 2024] And that is exactly what we want to do.

Disposition: Rejected

Specialization hierarchy

The <glossdef> element is specialized from <abstract>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [universal attributes](#) (51).

Example

The following code sample shows the <glossdef> element can be used to define the meaning of the term "raster pattern":

```
<glossentry id="rasterpattern">
  <glossterm>raster pattern</glossterm>
  <glossdef>A series of picture elements (pels) arranged in scan lines to form
    an image.</glossdef>
</glossentry>
```

3.7 <glossentry>

The <glossentry> element is the top-level element for a topic that defines a glossary term.

Rendering expectations

Because the glossary entry specialization is designed for the larger task of terminology management, it contains many elements that are not intended to be rendered when a glossary is generated.

Comment by Zoë Lawson

Do we want to indicate which elements are not intended to be rendered "by default"? (Not that you can't customize anything you want.) Not that I've really dug into glossaries, but If I'm just trying to figure out what to do reading the spec, I'd have zero idea which elements probably aren't emitted by the DITA-OT.

[Kris Eberlein, 24 March 2024] Zoe, that would be useful information for a book about the DITA-OT or a company styleguide, but it's not what we want in the DITA specification.

Disposition: Closed

In addition, when a collections

Comment by Robert Johnson

"...when a collection of glossary entry topics is rendered..."

[Kris Eberlein, 24 March 2024] Good catch. Done.

Disposition: Completed

of glossary entry topics are rendered as a guidance for terminology management, generated text might be required for ease of reading. Specialized style sheets and processing are needed to ensure useful output.

Processing expectations

Processing expectations for glossary entry topics are highly implementation-specific and will depend on the output format.

The typical processing for books is to sort and group glossary entries based on the localized term, so a back-of-the-book glossary can contain a collated list of terms with the definitions of the individual senses of each term indented under the term. The glossary can have a different organization in different languages depending on the translation of the terms.

Comment by dawnstevens

The phrasing of the first sentence suggests to me that we expect the processor to automatically sort the terms alphabetically. Should we rephrase?

[Kris Eberlein, 24 March 2024] I rephrased as Robert suggested AND moved the content into the localization topic.

Disposition: Completed**Comment by robander on Jan 18 2024**

Agreed, I don't know of tools that do that now and the spec should not imply that it is "expected" – that said it is certainly a valid implementation, so it could be described like the next paragraph, "one possible implementation"

WEK: I implemented an OT plug-in to do exactly this kind of locale-aware grouping and sorting of glossary entries (and other things that need grouping and sorting) but I never got client funding to finish it. ServiceNow is now both publishing a glossary and planning to localize it, so maybe I will be directed to finish that work at long last...

That said, per my other comments on l10n discussion: I think all localization discussion should be moved to a dedicated topic or separate white paper.

[Kris Eberlein, 24 March 2024] Moved the content to the new "<abbreviated-form> element and localization" topic.

Disposition: Completed

One possible implementation of a glossary in online processing is to associate a hotspot for mentions of terms in <term> elements and display the definition on hover or click. A possible implementation in PDF processing is to define a first-use occurrence form of the term with its acronym and use it in conjunction

with `<abbreviated-form>` to render a spell-out version of the term with its acronym on first use within an output, while rendering the acronym in all subsequent locations.

Specialization hierarchy

The `<glossentry>` element is specialized from `<concept>`; it is defined in the glossary entry module. The `<concept>` element is specialized from `<topic>`; it is defined in the concept module.

Attributes

The following attributes are available on this element: [architectural attributes](#) (56) and [universal attributes](#) (51).

For this element, the `@id` attribute is required.

Example

The following code samples shows how a glossary entry topic provides information about a term that aids in terminology management:

```
<glossentry id="usbfd">
  <glossterm>USB flash drive</glossterm>
  <glossdef>A small portable drive.</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossUsage>Do not use this term in upper case (for example, in "USB Flash Drive")
      because that suggests a trademark.</glossUsage>
    <glossAlt>
      <glossAcronym>UFD</glossAcronym>
      <glossStatus value="prohibited"/>
    </glossAlt>
    <glossAlt id="memoryStick">
      <glossSynonym>memory stick</glossSynonym>
      <glossUsage>This is a colloquial term.</glossUsage>
    </glossAlt>
    <glossAlt>
      <glossAbbreviation>stick</glossAbbreviation>
      <glossStatus value="prohibited"/>
      <glossUsage>This is too colloquial.</glossUsage>
      <glossAlternateFor href="#usbfd/memoryStick"/>
    </glossAlt>
    <glossAlt>
      <glossAbbreviation>flash</glossAbbreviation>
      <glossStatus value="prohibited"/>
      <glossUsage>This short form is ambiguous.</glossUsage>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.8 <glossPartOfSpeech>

The `<glossPartOfSpeech>` element identifies the part of speech for the term specified in the topic's `<glossterm>` element, and any variant forms of the term also defined within the topic. term and any alternate or variant forms of the term defined within the topic.

Comment by Eliot Kimber on 2024-01-28

Something wrong with the content here--duplicated words?

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Usage information

The part of speech is defined within the @value attribute. If validation is required, use a subject scheme to configure controlled values for the @value attribute on the <glossPartOfSpeech>. This approach avoids the need to hard code values in the XML grammar files, which would limit extension and would make values difficult to change for different languages.

When the part of speech is not specified, the default is "noun."

Comment by dawnstevens

why is there a default?

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Comment by dawnstevens

why isn't there a defined list of values? seems like we have a pre-defined list of possible values.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Comment by Robert Johnson

I agree. The set of parts of speech is pretty well defined.

WEK: Based on this Wikipedia article, parts of speech are not universal: https://en.wikipedia.org/wiki/Part_of_speech. But we do define starter sets of enumerated values for other attributes, seems appropriate to do so here.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Specialization hierarchy

The <glossPartOfSpeech> element is specialized from <data>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [data-element attributes](#) (56), [link-relationship attributes](#) (56), and [universal attributes](#) (51).

For this element, the @value attribute specifies the part of speech for the term or terms.

Example

The following code sample shows how the <glossPartOfSpeech> element can be used to specify the part of a speech for a term:

```
<glossentry id="refridgerator">
  <glossterm>Refrigerator</glossterm>
  <glossdef>A refrigerator is a commercial and home appliance consisting
    of a thermally insulated compartment and a heat pump (mechanical,
```



```

    electronic or chemical) that transfers heat from its inside to its
    external environment so that its inside is cooled to a temperature
    below the room temperature.</glossdef>
<glossBody>
  <glossPartOfSpeech value="noun"/>
  <glossAlt>
    <glossSynonym>fridge</glossSynonym>
    <glossUsage>Casual and colloquial usage only.</glossUsage>
  </glossAlt>
</glossBody>
</glossentry>

```

3.9 <glossProperty>

The <glossProperty> element specifies additional details about the term specified in the topic's <glossterm> element. For example, this element might be specified to specify the gender of a term in languages that have such a distinction or it might provide the term's etymological origins.

Comment by Robert Johnson

"...this element might be included to specify the gender..."

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Usage information

The specific type of detail being defined is specified in the @name attribute and its corresponding value in the @value attribute. Because any number of properties can therefore be defined, implementers may want to control these values using subject scheme maps to bind values to these attributes within <glossProperty>.

Comment by Eliot Kimber on 2024-01-28

or create specializations of <glossProperty> for specific properties.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Specialization hierarchy

The <glossProperty> element is specialized from <data>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [data-element attributes](#) (56), [link-relationship attributes](#) (56), and [universal attributes](#) (51).

Example

The following code sample shows how the <glossProperty> element can be used to provide information about the gender of a term:

```

<glossentry id="algorithm" xml:lang="es-es">
  <glossterm>El algoritmo</glossterm>
  <glossdef>Un algoritmo define un método de calcular un resultado.</glossdef>
  <glossBody>

```

```

    <glossPartOfSpeech value="noun"/>
    <glossProperty name="gender" value="masculine"/>
  </glossBody>
</glossentry>

```

The following code sample shows how the `<glossProperty>` element can be used to capture the part number and OEM for equipment or parts defined in a glossary term:

```

<glossentry id="anvil">
  <glossterm>anvil</glossterm>
  <glossdef>A heavy steel or iron block with a flat top, concave sides, and pointed end, on which
    metal can be hammered and shaped. Handy for dropping on roadrunners.</glossdef>
  <glossBody>
    <glossProperty name="part_number" value="123456"/>
    <glossProperty name="eom" value="Acme Company"/>
  </glossBody>
</glossentry>

```

3.10 <glossScopeNote>

The `<glossScopeNote>` element defines the extent to which or context within which the term specified in the topic's `<glossterm>` element might be used.

Specialization hierarchy

The `<glossScopeNote>` element is specialized from `<note>`. It is defined in the glossary entry module.

Comment by dawnstevens

Although I realize the element is specialized from Note, I question the need to point out the type and othertype attributes; they don't seem to me to have any meaning in the content of a scope note. A scope note is type scope note, in my mind. Why would anyone specify any type or othertype?

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Comment by dawnstevens

While it is true that type and othertype are valid attributes because `glossScopeNote` is specialized from Note, I don't think there is any meaning for a `glossScopeNote` to be any specific "type". Its type is a scope note. Do we have to document that these attributes are available?

[Robert Anderson, 18 January 2024] For every attribute that is defined, we need to include it in the attribute list, whether we expect it to be used or not. If we want to modify this (but not remove the attributes), then we need to modify the conref to "note" so that we pull in the attribute list, but then add an exception saying that these attributes will not typically be used, that the type is expected to be a scope note.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

]

Comment by Eliot Kimber on 2024-01-28

Given this is a specialization, why not just set @type to "other" and then default the value of @othertype to "scopenote"?

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Attributes

The following attributes are available on this element: [universal attributes](#) (51) and the attributes defined below.

@othertype

Specifies an alternate note type. This value is used as the user-provided note label when the @type attribute value is set to "other".

@type

Specifies the type of a note. This differs from the @type attribute on many other DITA elements. The following are the allowable values:

- "attention"
- "caution"
- "danger"
- "important"
- "note"
- "notice"
- "other"
- "remember"
- "restriction"
- "tip"
- "trouble"
- "warning"
- "-dita-use-conref-target"

Example

The following code sample shows how the <glossScopeNote> element can be used to limit the scope in which the term "Linux" should be used:

```
<glossentry id="linuxOS" xml:lang="en-us">
  <glossterm>Linux operating system</glossterm>
  <glossdef>An operating system based on the kernel created by Linus Torvald.</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossScopeNote>This term does not apply to the Linux Open Source Project that
      develops Linux distributions, but only to the distributions themselves. Examples
      of Linux operating systems include RedHat, SuSE, and Ubuntu. </glossScopeNote>
  </glossBody>
</glossentry>
```

3.11 <glossShortForm>

The <glossShortForm> element is a form of the term specified in the topic's <glossterm> element that includes fewer words than the full form.

Usage information

Note Any list of alternative terms is specific to the language. Translation of a <glossentry> topic might result in an empty <glossShortForm> element if there is no equivalent short form in the target language.

Comment by Eliot Kimber on 2024-01-28
More l10n discussion.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Specialization hierarchy

The <glossShortForm> element is specialized from <title>. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [ID and conref attributes](#) (51), [localization attributes](#) (52), [@base](#) (52), [@class](#) (52), [@outputclass](#) (54), and [@rev](#) (55).

Example

The following code sample shows how the <glossShortForm> can be used.

```
<glossentry id="www">
  <glossterm>World Wide Web</glossterm>
  <glossdef>A system of interconnected web pages accessible through the Internet.</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossAlt>
      <glossShortForm>the web</glossShortForm>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.12 <glossStatus>

The <glossStatus> element identifies the status of the term specified in the topic's <glossterm> element or any of its alternative forms specified within a <glossAlt> element.

Comment by Eliot Kimber on 2024-01-28

Neither the short description nor usage really explain what "status" means in this context, although the example does indicate that it means "the status of the term as a part of a controlled or managed vocabulary."

It would just as easily mean "the status of the term in the context of a terminology development workflow".

It seems likely that only the first meaning is intended.

[Kris Eberlein, 05 March 2024] Marking this comment as "Closed" since we are removing the element.

Disposition: Closed

Usage information

Status is specified in the `@value` attribute. If validation is required for status values, use a subject scheme to set up controlled values for the `@value` attribute. Alternatively, processing rules can be used to validate the values.

Specialization hierarchy

The `<glossStatus>` element is specialized from `<data>`. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [data-element attributes](#) (56), [link-relationship attributes](#) (56), and [universal attributes](#) (51).

For this element, the `@value` attribute specifies the usage status of the term or alternate term.

Example

The following code sample shows how the `<glossStatus>` element identifies the usage status of the glossary term and its alternates:

```
<glossentry id="twitter">
  <glossterm>X</glossterm>
  <glossdef>A social media platform</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossStatus value="preferred"/>
    <glossAlt>
      <glossSynonym>Twitter</glossSynonym>
      <glossStatus value="out-of-date"/>
      <glossUsage>Use in metadata tags to direct individuals who still use this term to the
appropriate location.</glossUsage>
    </glossAlt>
    <glossAlt>
      <glossSynonym>Twitter/X</glossSynonym>
      <glossStatus value="prohibited"/>
      <glossUsage>Use instead "X, formerly known as Twitter".</glossUsage>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.13 <glossSurfaceForm>

The `<glossSurfaceForm>` element specifies how the term specified by the topic's `<glossterm>` element should appear in the text. The surface form is suitable to introduce the term in new contexts or as the first occurrence in text.

Comment by dawnstevens

I learned a lot more than I really wanted to about the surface form in linguistics to suggest this short description, as opposed to what was there: "an unambiguous presentation of the term that might combine multiple forms." Surface forms are "the way a linguistic unit (such as a word, phrase, or sentence) appears when it is actually spoken or written, as opposed to its underlying or abstract form."

Surface forms may take into account phonology (to provide stress, intonation, and accent), test and plurals (the surface form of "go" is "goes" in 3rd person singular), and derivational (to change roots of words in various contexts), contextual (I'm might be a surface form of I am in informal speech). Ultimately, in my experience the only thing this is really used for is first-use occurrence of acronyms so they are spelled out. I wonder if we should use that use case only in this definition, rather than a more comprehensive scope.

WEK: Your more technical definition implies that a given abstract term might have multiple surface forms depending on usage context (for example, some language's grammar might require different surface forms depending on whether a noun is the subject or object of verb). Given that you're allowed at most one surface form per glossary entry, that seems pretty limiting.

Maybe a better element type name is "glossFirstUseForm" or something?

Or, if we really need to allow multiple surface forms, it should be one of the allowed titles of `<glossAlt>`. Or maybe we need both things...

[Kris Eberlein, 24 March 2024] Marking this comment closed, and leaving the short description largely as-is (I did add a "syntactic that"), given our decision to pare back the number of elements in the specialization.

Disposition: Closed

Usage information

The `<glossSurfaceForm>` element is most often used for terms that also specify the `<glossAcronym>` element. In that context, the `<glossSurfaceForm>` element contains the term in a manner that introduces both the term and the acronym, so that later references to the term can be replaced with the acronym alone.

Note In some languages, the surface form that expands the acronym in its first use handles the formatting differently than in English. For example, in Polish, the acronym precedes the expansion. In addition, some languages may not have a corresponding acronym for a term, so the `<glossSurfaceForm>` and `<glossAcronym>` would be left blank in translated content.

See the `<abbreviated-form>` element for a full description of how the surface form is used together with acronyms.

Specialization hierarchy

The `<glossSurfaceForm>` element is specialized from `<p>`. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [universal attributes](#) (51).

Example

The following glossary entry topic defines the term "Anti-lock Braking System". Within the topic, the `<glossSurfaceForm>` element provides a version of the term that combines both the primary term and the acronym. The content of the `<glossSurfaceForm>` might be rendered in introductory contexts when the glossary entry topic is referenced from an `<abbreviated-form>` element.

```
<glossentry id="abs">
  <glossterm>Anti-lock Braking System</glossterm>
```

```

<glossBody>
  <glossSurfaceForm>Anti-lock Braking System (ABS)</glossSurfaceForm>
  <glossAlt>
    <glossAcronym>ABS</glossAcronym>
  </glossAlt>
</glossBody>
</glossentry>

```

3.14 <glossSymbol>

The <glossSymbol> element identifies a standard image that is associated with the subject of the term specified by the topic's <glossterm> element. For example, the image might serve to visually classify the term into a specific taxonomy.

Comment by Eliot Kimber on 2024-01-28

Why is this association of an image with a term privileged in this way? It doesn't make any obvious sense on its face to me and the short description doesn't help, as the example of a visual classifier seems odd, or at least, highly use case specific.

The original design must have had some use cases in mind but visual classification seems very specific, something you might handle in your own domain-specific specialization of glossAlt.

ZLL: While I understand it could be 'easily' specialized, as someone who's never made a specialization, having something like this available that "works out of the box" seems great.

[Kris Eberlein, 24 March 2024] Removed the second sentence in the short description.

Disposition: Completed

Specialization hierarchy

The <glossSymbol> element is specialized from <image>. It is defined in the glossary entry module.

Comment by Christina Rothwell on January 23, 2024

Parts of sentences appear as red text in the PDF in the @height, @scalefit, and @width sections. Why is the text red?

ZLL: If I remember correctly, there are some @rev that highlight changes

[Kris Eberlein, 05 March 2024] Zoe is correct. If content was revised as a result of an earlier review, it is marked using the @revattribute. Our style sheets render content marked with in red.

Disposition: Closed

Attributes

The following attributes are available on this element: [universal attributes](#) (51), [@format](#) (61), [@href](#) (61), [@keyref](#) (62), [@scope](#) (66), and the attributes defined below.

@align

Controls the horizontal alignment of an image when @placement is specified as "break". Common values include "left", "right", and "center".

@height

Specifies the vertical dimension for the resulting display. The value of this attribute is a real number expressed in decimal notation, optionally followed by a unit of measure. The following units of measurement are supported: cm, em, in, mm, pc, pt, and px (centimeters, ems, inches, millimeters, picas, points, and pixels, respectively). The default unit is px (pixels). Possible values include: "5", "5in", and "10.5cm".

@placement

Indicates whether an image is displayed inline or on a separate line. The default value is inline. Allowable values are "inline", "break", and "-dita-use-conref-target" (68).

@scale

Specifies a percentage as an unsigned integer by which to scale the image in the absence of any specified image height or width; a value of 100 implies that the image should be presented at its intrinsic size. If a value has been specified for the @height or @width attribute (or both), the @scale attribute is ignored.

@scalefit

Specifies whether an image is scaled up or down to fit within available space. The allowable values are "yes", "no", and "-dita-use-conref-target" (68). If @height, @width, or @scale is specified, those attributes determine the graphic size, and the @scalefit attribute is ignored. If none of those attributes are specified and scalefit="yes", then the image is scaled by the same factor in both dimensions, so that the graphic will just fit within the available height or width, whichever is more constraining.

The available width would be that of the prevailing column or table cell, that is, the width a paragraph of text would have if the graphic were a paragraph instead of text. The available height is implementation dependent, but if feasible, it is suggested to be the page or table cell height or some other reasonable value.

@width

Specifies the horizontal dimension for the resulting display. The value of this attribute is a real number expressed in decimal notation, optionally followed by a unit of measure. The following units of measurement are supported: cm, em, in, mm, pc, pt, and px (centimeters, ems, inches, millimeters, picas, points, and pixels, respectively). The default unit is px (pixels). Possible values include: "5", "5in", and "10.5cm".

Example

The following code sample shows how the <glossSymbol> can be used to associate a regional classification icon with the Atlantic puffin:

```
<glossentry id="atlanticpuffin">
  <glossterm>Atlantic Puffin</glossterm>
  <glossdef>A sea bird that lives in the Atlantic
    <image href="puffinicon.jpg">
      <alt>Image of an atlantic puffin</alt>
    </image>
  </glossdef>
  <glossBody>
    <glossSymbol href="atlantic.jpg" scope="local">
      <alt>Icon denoting the Atlantic region</alt>
    </glossSymbol>
  </glossBody>
</glossentry>
```

3.15 <glossSynonym>

The <glossSynonym> element provides a term that is a synonym of the term specified by the topic's <glossterm> element, but that is not an acronym, abbreviation, or short form.

Comment by Eliot Kimber on 2024-01-28

I think this element is the answer to Dawn's question about `<glossAlternateFor>` not applying to the entire glossary definition. This element serves that role and, because it allows both `<keyword>` and `<term>`, can be a link to the glossary entry of the synonym term.

[Kris Eberlein, 24 March 2024] Added Eliot's comment to the source file; Dawn can decide whether to elaborate on this point in the current example or in an additional example.

Disposition: Completed

Usage information

Note A list of synonyms is specific to the language, so translation of a `<glossentry>` topic might result in empty `<glossSynonym>` elements.

Specialization hierarchy

The `<glossSynonym>` element is specialized from `<title>`. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [ID and conref attributes](#) (51), [localization attributes](#) (52), [@base](#) (52), [@class](#) (52), [@outputclass](#) (54), and [@rev](#) (55).

Example

The following code sample shows how the `<glossSynonym>` element can be used to identify a synonym for the word automobile:

```
<glossentry id="automobile">
  <glossterm>Automobile</glossterm>
  <glossdef>A road vehicle, typically with four wheels, powered by an
internal combustion engine or an electric motor.</glossdef>
  <glossBody>
    <glossAlt>
      <glossSynonym>car</glossSynonym>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.16 <glossterm>

The `<glossterm>` element specifies the term addressed

Comment by Eliot Kimber

"addressed within" implies addresssing in the linking sense. I think "term defined by the `<glossentry>` topic" would be clearer.

[Kris Eberlein, 05 March 2024] Agree. We should be sure to make the same wording change in the introduction to the example.

[Kris Eberlein, 24 March 2024] Changed to read as follows:

The `<glossterm>` element specifies the term that is defined by the glossary entry topic.

Disposition: Completed

within the `<glossentry>` topic.

Specialization hierarchy

The `<glossterm>` element is specialized from `<title>`. It is defined in the glossary entry module.

Attributes

The following attributes are available on this element: [ID and conref attributes](#) (51), [localization attributes](#) (52), [@base](#) (52), [@class](#) (52), [@outputclass](#) (54), and [@rev](#) (55).

Example

The following code sample shows how the `<glossterm>` element specifies the term addressed within the `<glossentry>` topic:

```
<glossentry id="css">
  <glossterm>cascading style sheets</glossterm>
  <glossdef>Cascading Style Sheets is a style sheet language that is used
    for rendering the presentation of a document written in a markup
    language such as HTML or XML</glossdef>
  <glossBody>
    <glossSurfaceForm>cascading style sheets (CSS)</glossSurfaceForm>
    <glossAlt>
      <glossAcronym>CSS</glossAcronym>
    </glossAlt>
    <glossAlt>
      <glossSynonym>web style sheets</glossSynonym>
    </glossAlt>
  </glossBody>
</glossentry>
```

3.17 <glossUsage>

The `<glossUsage>` element provides information about how to use the term specified in the topic's `<glossterm>` element or any of its alternative forms specified within a `<glossAlt>` element.

Comment by Eliot Kimber on 2024-01-28

"specified within `<glossAlt>` elements"

[Kris Eberlein, 05 March 2024] I think we should leave the rewording to Dawn Stevens as she reworks the content.

Disposition: Closed

Specialization hierarchy

The `<glossUsage>` element is specialized from `<note>`. It is defined in the glossary entry module.

Comment by dawnstevens

Although I realize the element is specialized from Note, I question the need to point out the type and othertype attributes; they don't seem to me to have any meaning in the content of a usage note. A usage note is type usage note, in my mind. Why would anyone specify any type or othertype?

[Kris Eberlein, 24 March 2024] Commenting out the draft comment present in the source.

Disposition: Closed

Comment by robander on Jan 18, 2024

For every attribute that is defined, we need to include it in the attribute list, whether we expect it to be used or not. If we want to modify this (but not remove the attributes), then we need to modify the conref to "note" so that we pull in the attribute list, but then add an exception saying that these attributes will not typically be used, that the type is expected to be a scope usage note.

[Kris Eberlein, 24 March 2024] Commenting out the draft comment present in the source.

WEK: As for the other note specialization, we should just set the @type value and the @othertype value.

Disposition: Closed

Comment by Tony Self

Further to Robert's note, the (no value) option is sometimes defined in the attribute list. For example, the Spec provides (no value) information for the type attribute on linking elements. It is not defined for <note> (and perhaps it is not needed), but it might be more relevant for specialisations of <note>.

[Kris Eberlein, 24 March 2024] Marking this comment as "Closed"; no action to be taken.

Disposition: Closed

Attributes

The following attributes are available on this element: [universal attributes](#) (51) and the attributes defined below.

@othertype

Specifies an alternate note type. This value is used as the user-provided note label when the @type attribute value is set to "other".

@type

Specifies the type of a note. This differs from the @type attribute on many other DITA elements. The following are the allowable values:

- "attention"
- "caution"
- "danger"
- "important"
- "note"
- "notice"
- "other"
- "remember"
- "restriction"
- "tip"
- "trouble"
- "warning"
- "-dita-use-conref-target"

Example

The following code sample shows how the `<glossUsage>` element is used to provide additional information about possible variants for the term "soft drink:"

```
<glossentry id="softdrink">
  <glossterm>soft drink</glossterm>
  <glossdef>A nonalcoholic drink, especially one that is carbonated.</glossdef>
  <glossBody>
    <glossPartOfSpeech value="noun"/>
    <glossAlt>
      <glossSynonym>pop</glossSynonym>
      <b><glossUsage>Used primarily in the North and Midwest</glossUsage></b>
    </glossAlt>
    <glossAlt>
      <glossSynonym>soda</glossSynonym>
      <b><glossUsage>Used primarily in the West and Northeast</glossUsage></b>
    </glossAlt>
    <glossAlt>
      <glossSynonym>Coke</glossSynonym>
      <b><glossUsage>Used primarily in the South</glossUsage></b>
    </glossAlt>
  </glossBody>
</glossentry>
```

A Aggregated RFC-2119 statements

This appendix contains all the normative statements from the DITA for Technical Content 2.0 specification. They are aggregated here for convenience in this non-normative appendix.

B Attributes

This section contains definitions for commonly-used attributes. If an attribute is defined differently on a specific element, that information is covered in the topic for the specific element.

Comment by Kristen J Eberlein on 29 December 2021

Add a brief overview of the fact that some specific attributes are overloaded – and have different meanings depending on what element they are specified upon.

Disposition: Unassigned

B.1 Attribute groups

Many of the attributes used on DITA elements are defined in attribute groups. These attribute groups are used both in the grammar files and the specification,

Architectural attributes

This group contains a set of attributes that are defined for document-level elements such as `<topic>` and `<map>`.

@DITAArchVersion (architectural attributes)

Specifies the version of the DITA architecture that is in use. This attribute is in the namespace `http://dita.oasis-open.org/architecture/2005/`. This attribute is specified in the topic and map modules, and it uses a default value of the current version of DITA. The current default is "2.0".

@specializations (architectural attributes)

Specifies the attribute-domain specializations that are included in the document-type shell. This attribute is set as a default within the document-type shell. The value varies depending on what domains are integrated into the document-type shell. For example, a grammar file that includes the specialized attributes `@audience`, `@deliveryTarget`, and `@newBaseAtt` would set the value to `@props/audience @props/deliveryTarget @base/newBaseAtt`.

@xmlns:ditaarch (architectural attributes)

Declares the default DITA namespace. This namespace is declared as such in the RNG modules for `<topic>` and `<map>`, but it is specified as an attribute in the equivalent DTD-based modules. The value is fixed to `"http://dita.oasis-open.org/architecture/2005/"`.

Common map attributes

This group contains attributes that are frequently used on map elements.

Comment by Kristen J Eberlein on 28 September 2022

I've added draft comments to the attribute definitions in this section that explain how the attribute is defined in the "DITA map attributes" topic.

Disposition: Unassigned

@cascade (common map attributes)

Specifies how metadata attributes cascade within a map. The specification defines the following values:

merge

Indicates that the metadata attributes cascade, and that the values of the metadata attributes are additive. This is the processing default for the `@cascade` attribute.

nomerge

Indicates that the metadata attributes cascade, but that they are not additive for `<topicref>` elements that specify a different value for a specific metadata attribute. If the cascading value for an attribute is already merged based on multiple ancestor elements, that merged value continues to cascade until a new value is encountered. That is, setting `cascade="nomerge"` does not undo merging that took place on ancestor elements.

Processors can also define custom, implementation-specific tokens for this attribute.

See [Cascading of metadata attributes in a DITA map](#) for more information about how this attribute interacts with metadata attributes.

@chunk (common map attributes)

Specifies how a processor should render a map or branch of a map. For example, it can be used to specify that individual topic documents should be rendered as a single document, or that a single document with multiple topics should be rendered as multiple documents.

The following values are valid:

combine

Instructs a processor to combine the referenced source documents for rendering purposes. This is intended for cases where a publishing process normally results in a single output artifact for each source XML document.

split

Instructs a processor to split each topic from the referenced source document into its own document for rendering purposes. This is intended for cases where a publishing process normally results in a single output artifact for each source XML document, regardless of how many DITA topics exist within each source document.

Processors can also define custom, implementation-specific tokens for this attribute.

For a detailed description of the `@chunk` attribute and its usage, see [Chunking](#).

@collection-type (common map attributes)

Specifies how topics or links relate to each other. The processing default is "unordered", although no default is specified in the OASIS-provided grammar files. The following values are valid:

unordered

Indicates that the order of the child topics is not significant.

sequence

Indicates that the order of the child topics is significant. Output processors will typically link between them in order.

choice

Indicates that one of the children should be selected.

family

Indicates a tight grouping in which each of the referenced topics not only relates to the current topic but also relate to each other.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@collection-type

The `@collection-type` attribute specifies how the children of a `<topicref>` element relate to their parent and to each other. This attribute, which is set on the parent element,

typically is used by processors to determine how to generate navigation links in the rendered topics. For example, a `@collection-type` value of "sequence" indicates that children of the specifying `<topicref>` element represent an ordered sequence of topics; processors might add numbers to the list of child topics or generate next/previous links for online presentation. This attribute is available in topics on the `<linklist>` and `<linkpool>` elements, where it has the same behavior. Where the `@collection-type` attribute is available on elements that cannot directly contain elements, the behavior of the attribute is undefined.

Disposition: Unassigned

Comment by Kristen J Eberlein on 28 September 2022

In the definitions of the supported values, do we want to refer to "resources" instead of "topics"? Since we specify that `@collection-type` specifies "how topics **or** links relate to each other" ...

Disposition: Unassigned

@keyscope (common map attributes)

Specifies that the element marks the boundaries of a key scope.

See [B.4 STUB CONTENT](#) (68) for information on using this attribute.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@keyscope

Defines a new scope for key definition and resolution, and gives the scope one or more names. For more information about key scopes, see [Indirect key-based addressing](#).

Disposition: Unassigned

@linking (common map attributes)

Specifies linking characteristics of a topic specific to the location of this reference in a map. If the value is not specified locally, the value might cascade from another element in the map (for cascade rules, see [Cascading of metadata attributes in a DITA map](#)).

Comment by robander on Dec 28 2021

The text below matches [1.3 spec text](#) but I'm nervous about "cannot link" type definition. It's describing how to generate links based on the current context in the map - it's not describing what the topic itself is allowed to link to, which is how I interpret "can".

Disposition: Unassigned

The following values are valid:

targetonly

A topic can only be linked to and cannot link to other topics.

sourceonly

A topic cannot be linked to but can link to other topics.

normal

A topic can be linked to and can link to other topics. Use this to override the linking value of a parent topic.

none

A topic cannot be linked to or link to other topics.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@linking

By default, the relationships between the topics that are referenced in a map are reciprocal:

- Child topics link to parent topics and vice versa.
- Next and previous topics in a sequence link to each other.
- Topics in a family link to their sibling topics.
- Topics referenced in the table cells of the same row in a relationship table link to each other. A topic referenced within a table cell does not (by default) link to other topics referenced in the same table cell.

This behavior can be modified by using the `@linking` attribute, which enables an author or information architect to specify how a topic participates in a relationship. The following values are valid:

linking="none"

Specifies that the topic does not exist in the map for the purposes of calculating links.

linking="sourceonly"

Specifies that the topic will link to its related topics but not vice versa.

linking="targetonly"

Specifies that the related topics will link to it but not vice versa.

linking="normal"

Default value. It specifies that linking will be reciprocal (the topic will link to related topics, and they will link back to it).

Authors also can create links directly in a topic by using the `<xref>` or `<link>` elements, but in most cases map-based linking is preferable, because links in topics create dependencies between topics that can hinder reuse.

Note that while the relationships between the topics that are referenced in a map are reciprocal, the relationships merely *imply* reciprocal links in generated output that includes links. The rendered navigation links are a function of the presentation style that is determined by the processor.

Disposition: Unassigned

@processing-role (common map attributes)

Specifies whether the referenced resource is processed normally or treated as a resource that is only included in order to resolve references, such as key or content references. The following values are valid:

normal

Indicates that the resource is a readable part of the information set. It is included in navigation and search results. This is the default value for the `<topicref>` element.

resource-only

Indicates that the resource should be used only for processing purposes. It is not included in navigation or search results, nor is it rendered as a topic. This is the default value for the `<keydef>` element.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

@search (common map attributes)

Specifies whether the target is available for searching. If the value is not specified locally, the value might cascade from another element in the map (for cascade rules, see [Cascading of metadata attributes in a DITA map](#)). The following values are valid: "yes", "no", and "-dita-use-conref-target".

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@search

Specifies whether the topic is included in search indexes.

Disposition: Unassigned

@subjectrefs (common map attributes)

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.

@toc (common map attributes)

Specifies whether a topic appears in the table of contents (TOC) based on the current map context. If the value is not specified locally, the value might cascade from another element in the map (for cascade rules, see [Cascading of metadata attributes in a DITA map](#)). The following values are valid:

yes

The topic appears in a generated TOC.

no

The topic does not appear in a generated TOC.

-dita-use-conref-target

See [B.4 STUB CONTENT](#) (68) for more information.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@toc

Specifies whether topics are excluded from navigation output, such as a Web site map or an online table of contents. By default, <topicref> hierarchies are included in navigation output; relationship tables are excluded.

Disposition: Unassigned

Complex table attributes

This group includes attributes that are defined on complex table elements. Unless other noted, these attributes are part of the OASIS Exchange Table Model. Complex table elements typically use only a subset of the attributes that are defined in this group.

@align (complex table attributes)

Specifies the **horizontal** alignment of text in table **entries**. The following values are valid:

left

Indicates left alignment of the text.

right

Indicates right alignment of the text.

center

Indicates center alignment of the text.

justify

Justifies the contents to both the left and the right.

char

Indicates character alignment. The text is aligned with the first occurrence of the character specified by the `@char` attribute.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

The `@align` attribute is available on the following table elements: `<colspec>`, `<entry>`, and `<tgroup>`.

@char (complex table attributes)

Specifies the alignment character, which is the character that is used for aligning the text in table entries. This attribute applies when `align="char"`. A value of `""` (the null string) means there is no aligning character.

For example, if `align="char"` and `char="."` are specified, then text in the table entry aligns with the first occurrence of the period within the entry. This might be useful if decimal alignment is required.

The `@char` attribute is available on the following table elements: `<colspec>` and `<entry>`.

@charoff (complex table attributes)

Specifies the horizontal offset of the alignment character **that is specified by the `@char` attribute**. The value is a greater-than-zero number that is less than or equal to 100. It represents the percentage of the current column width by which the text is offset to the left of the alignment character.

For example, if `align="char"`, `char="."`, and `charoff="50"` are all specified, then text in the table entry is aligned 50% of the distance to the left of the first occurrence of the period character within the table entry.

The `@charoff` attribute is available on the following table elements: `<colspec>` and `<entry>`.

@colsep (complex table attributes)

Specifies whether to render column separators between table entries. The following values are valid: `"0"` (no separators) and `"1"` (separators).

The `@colsep` attribute is available on the following table elements: `<colspec>`, `<entry>`, `<table>`, and `<tgroup>`.

@rowheader (complex table attributes)

Specifies whether the entries in the respective column are row headers. The following values are valid:

firstcol

Indicates that entries in the first column of the table are row headers. This applies when the `@rowheader` attribute is specified on the `<table>` element.

headers

Indicates that entries of the column that is described using the `<colspec>` element are row headers. This applies when the `@rowheader` attribute is specified on the `<colspec>` element.

norowheader

Indicates that entries in the first column are not row headers. This applies when the `@rowheader` attribute is specified on the `<table>` element.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Note This attribute is not part of the OASIS Exchange Table Model upon which DITA tables are based. Some processors or output formats might not support all values.

The `@rowheader` attribute is available on the following table elements: `<table>` and `<colspec>`.

@rowsep (complex table attributes)

Specifies whether to render row separators between table entries. The following values are valid: "0" (no separators) and "1" (separators).

The `@rowsep` attribute is available on the following table elements: `<colspec>`, `<entry>`, `<row>`, `<table>`, and `<tgroup>`.

@valign (complex table attributes)

Specifies the vertical alignment of text in table **entries**. The following values are valid:

bottom

Indicates that text is aligned with the bottom of the table entry.

middle

Indicates that text is aligned with the middle of the table entry.

top

Indicates that text is aligned with the top of the table entry.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

The `@valign` attribute is available on the following table elements: `<entry>`, `<tbody>`, `<thead>`, and `<row>`.

Data-element attributes

This group contains attributes that are defined on the `<data>` element and its specializations.

@datatype (data-element attributes)

Specifies the type of data contained in the `@value` attribute or within the `<data>` element. A typical use of `@datatype` will be the identifying URI for an XML Schema datatype.

@name (data-element attributes)

Defines a unique name for the object.

Comment by robander

Do we need to specify the scope of "unique" here?

Disposition: Unassigned

@value (data-element attributes)

Specifies a value associated with the current property or element.

Date attributes

This group contains attributes that take date values. They are defined on metadata elements that work with date information:

@expiry (date attributes)

Specifies the date when the information should be retired or refreshed. The date is specified using the ISO 8601 format: `YYYY-MM-DD`, where `YYYY` is the year, `MM` is the month (01 to 12), and `DD` is the day (01-31).

@golive (date attributes)

Specifies the publication or general availability (GA) date. The date is specified using the ISO 8601 format: `YYYY-MM-DD`, where `YYYY` is the year, `MM` is the month (01 to 12), and `DD` is the day (01-31).

Display attributes

This group contains attributes that affect the rendering of many elements.

@**expanse** (display attributes)

Specifies the horizontal placement of the element. The following values are valid:

column

Indicates that the element is aligned with the current column margin.

page

Indicates that the element is placed on the left page margin for left-to-right presentation or the right page margin for right-to-left presentation.

spread

Indicates that the object is rendered across a multi-page spread. If the output format does not have anything that corresponds to spreads, then "spread" has the same meaning as "page".

textline

Indicates that the element is aligned with the left (for left-to-right presentation) or right (for right-to-left presentation) margin of the current text line and takes indentation into account.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

For `<table>`, in place of the `@expanse` attribute that is used by other DITA elements, the `@pgwide` attribute is used in order to conform to the OASIS Exchange Table Model.

Some processors or output formats might not support all values.

@**frame** (display attributes)

Specifies which portion of a border surrounds the element. The following values are valid:

all

Indicates that a line is rendered at the top, bottom, left, and right of the containing element.

bottom

Indicates that a line is rendered at the bottom of the containing element.

none

Indicates that no lines are rendered.

sides

Indicates that a line is rendered at the left and right of the containing element.

top

Indicates that a line is rendered at the top of the containing element.

topbot

Indicates that a line is rendered at the top and bottom of the containing element.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Some processors or output formats might not support all values.

@**scale** (display attributes)

Specifies the percentage by which fonts are resized in relation to the normal text size. The value of this attribute is a positive integer. When used on `<table>` or `<simpletable>`, the following values are valid: "50", "60", "70", "80", "90", "100", "110", "120", "140", "160", "180", "200", and [-dita-use-conref-target](#) (68).

This attribute is primarily useful for print-oriented display. Some processors might not support all values.

If the `@scale` attribute is specified on an element that contains an image, the image is not scaled. The image is scaled **only** if a scaling property is explicitly specified for the `<image>` element.

ID and conref attributes

This group contains the attributes that enable the naming and referencing of elements.

@conaction

Specifies how the element content will be pushed into a new location. The following values are valid:

mark

The element acts as a marker when pushing content before or after the target, to help ensure that the push action is valid. The element with `conaction="mark"` also specifies the target of the push action with `@conref`. Content inside of the element with `conaction="mark"` is not pushed to the new location.

pushafter

Content from this element is pushed after the location specified by `@conref` on the element with `conaction="mark"`. The element with `conaction="pushafter"` is the first sibling element after the element with `conaction="mark"`.

pushbefore

Content from this element is pushed before the location specified by `@conref` on the element with `conaction="mark"`. The element with `conaction="pushbefore"` is the first sibling element before the element with `conaction="mark"`.

pushreplace

Content from this element replaces any content from the element referenced by the `@conref` attribute. A second element with `conaction="mark"` is not used when using `conaction="pushreplace"`.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

See [B.4 STUB CONTENT](#) (68) for examples and details about the syntax.

@conkeyref

Specifies a key name or a key name with an element ID that acts as an indirect reference to reusable content. The referenced content is used in place of the content of the current element. See [B.4 STUB CONTENT](#) (68) for more details about the syntax and behaviors.

@conref

Specifies a URI that references a DITA element. The referenced content is used in place of the content of the current element. See [B.4 STUB CONTENT](#) (68) for examples and details about the syntax.

@conrefend

Specifies a URI that references the last element in a sequence of elements, with the first element of the sequence specified by `@conref`. The referenced sequence of elements is used in place of the content of the current element. See [B.4 STUB CONTENT](#) (68) for examples and details about the syntax.

@id

Specifies an identifier for the current element. This ID is the target for references by `@href` and `@conref` attributes and for external applications that refer to DITA or LwDITA content. This attribute is defined with the XML data type NMTOKEN, except where noted for specific elements within the language reference.

See [id attribute](#) for more details.

Inclusion attributes

This group includes attributes defined on `<include>` and its specializations:

Comment by Kristen J Eberlein on 28 September 2002

What is specialized from `<include>`? Both base (if any) and technical content ...

Disposition: Unassigned

@encoding (inclusion attributes)

Comment by Kristen J Eberlein on 29 April 2019

Can we replace "should" in the following definition?

Disposition: Unassigned

Specifies the character encoding to use when translating the character data from the referenced content. The value should be a valid encoding name. If not specified, processors may make attempts to automatically determine the correct encoding, for example using HTTP headers, through analysis of the binary structure of the referenced data, or the `<?xml?>` processing instruction when including XML as text. The resource should be treated as UTF-8 if no other encoding information can be determined.

When `parse="xml"`, standard XML parsing rules apply for the detection of character encoding. The necessity and uses of `@encoding` for non-standard values of `@parse` are implementation-dependent.

@parse (inclusion attributes)

Specifies the processing expectations for the referenced resource. Processors must support the following values:

text

The contents should be treated as plain text. Reserved XML characters should be displayed, and not interpreted as XML markup.

xml

The contents of the referenced resource should be treated as an XML document, and the referenced element should be inserted at the location of the `<include>` element. If a fragment identifier is included in the address of the content, processors must select the element with the specified ID. If no fragment identifier is included, the root element of the referenced XML document is selected. Any grammar processing should be performed during resolution, such that default attribute values are explicitly populated. Prolog content must be discarded.

It is an error to use `parse="xml"` anywhere other than within `<foreign>` or a specialization thereof.

Processors may support other values for the `@parse` attribute with proprietary processing semantics. Processors should issue warnings and use `<fallback>` when they encounter unsupported `@parse` values. Non-standard `@parse` instructions should be expressed as URIs.

Note Proprietary `@parse` values will likely limit the portability and interoperability of DITA content, so should be used with care.

Link relationship attributes

This group contains attributes whose values can be used for representing navigational relationships.

@format (link-relationship attributes)

Specifies the format of the resource that is referenced. See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@href (link-relationship attributes)

Specifies a reference to a resource. See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@scope (link-relationship attributes)

Specifies the closeness of the relationship between the current document and the referenced resource. The following values are valid: "local", "peer", "external", and "-dita-use-conref-target".

See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@type (link-relationship attributes)

Describes the target of a reference. See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

Localization attributes

Comment by Kristen J Eberlein on 29 September 2022

The definition of the localizations attribute matches how they are described in the architectural topics. Wherever possible, the definition is reused. Where it is not reused (because the definition in the archSpec topics is in a shortdesc), I've checked to ensure that wording is identical.

Disposition: Unassigned

This group contains the attributes that are related to translation and localization.

@dir

Identifies or overrides the text directionality. The following values are valid:

lro

Indicates an override of the Unicode Bidirectional Algorithm, forcing the element into left-to-right mode.

ltr

Indicates left-to-right.

rlo

Indicates an override of the Unicode Bidirectional Algorithm, forcing the element into right-to-left mode.

rtl

Indicates right-to-left.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

See [The dir attribute](#) for more information.

@translate

Specifies whether the content of the element should be translated. The following values are valid: "yes", "no", and "-dita-use-conref-target".

See [Element-by-element recommendations for translators](#) for suggested processing defaults for each element.

Comment by Kristen J Eberlein on 31 December 2021

Does [Element-by-element recommendations for translators](#) really provide suggested processing defaults for each element? I thought it covered whether an element was block or in-line and whether there were considerations that translators needed to be aware of.

Disposition: Unassigned

@xml:lang

Specifies the language and optional locale of the content that is contained in an element. Valid values are language tokens or the null string. The `@xml:lang` attribute and its values are described in the [Extensible Markup Language 1.0 specification, fifth edition](#).

Comment by Kristen J Eberlein on 29 September 2022

Do we also want to direct readers to the architectural topics about the `@xml:lang` attribute?

Disposition: Unassigned

Metadata attributes

This group contains common metadata attributes: `@base`, `@importance`, `@props`, `@rev`, and `@status`. The `@base` and `@props` attributes can be specialized.

@base

Specifies metadata about the element. It is often used as a base for specialized attributes that have a simple syntax for values, but which are not conditional processing attributes.

The `@base` attribute takes a space-delimited set of values. However, when serving as a container for generalized attributes, the attribute values will be more complex. See [Attribute generalization](#) for more details.

@importance

Specifies the importance or priority that is assigned to an element. The following values are valid: "default", "deprecated", "high", "low", "normal", "obsolete", "optional", "recommended", "required", "urgent", and "-dita-use-conref-target". This attribute is not used for conditional processing, although applications might use the value of the `@importance` attribute to highlight elements. For example, in steps of a task topic, the value of the `@importance` attribute indicates whether a step is optional or required.

Comment by Kristen J Eberlein on 29 September 2022

I think the phrase "to highlight elements" is a little off. Maybe "render generated text"? And how about adding "Processors often add text or images to ensure that readers of the generated content understand whether the step is optional or required." to the end of the example?

Disposition: Unassigned

@props

Specifies metadata about the element. New attributes can be specialized from the `@props` attribute. This attribute supports conditional processing. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

The `@props` attribute takes a space-delimited set of values. However, when serving as a container for generalized attributes, the attribute values will be more complex. See [Attribute generalization](#) for more details.

@rev

Specifies a revision level of an element that identifies when the element was added or modified. It can be used to flag outputs when it matches a run-time parameter. It cannot be used for filtering nor is it sufficient to be used for version control. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

Comment by Kristen J Eberlein on 29 September 2022

I want to tweak this. How about the following? Also, neither definition describes what values are permitted.

Specifies metadata that identifies when the element was added or the content of the element was modified. The `@rev` attribute can be used for flagging. It cannot be used for filtering nor is it sufficient to be used for version control. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

Disposition: Unassigned

@status

Specifies the modification status of the element. The following values are valid: "new", "changed", "deleted", "unchanged", and "-dita-use-conref-target".

Simple table attributes

This group includes attributes that are defined only on the `<simpletable>` element: `@keycol` and `@relcolwidth`. These attributes are listed in a group because the `<simpletable>` element is frequently used as a specialization base.

@keycol (simpletable attributes)

Specifies the column that contains the content that represents the key to the tabular structure. If `@keycol` is present and assigned a numerical value, the specified column is treated as a vertical header.

@relcolwidth (simpletable attributes)

Specifies the width of each column in relationship to the width of the other columns. The value is a space-separated list of relative column widths. Each column width is specified as a positive integer or decimal number followed by an asterisk character.

For example, the value `relcolwidth="1* 2* 3*"` gives a total of 6 units across three columns. The relative widths are 1/6, 2/6, and 3/6 (16.7%, 33.3%, and 50%). Similarly, the value `relcolwidth="90* 150*"` causes relative widths of 90/240 and 150/240 (37.5% and 62.5%).

Table accessibility attributes

This group defines a set of attributes that promote table accessibility:

@headers

Specifies which entries in the current table provide headers for this cell. The `@headers` attribute contains an unordered set of unique, space-separated tokens, each of which is an ID reference of an entry from the same table.

@scope

Specifies that the current entry is a header for other table entries. The following values are valid:

col

Indicates that the current entry is a header for all cells in the column.

colgroup

Indicates that the current entry is a header for all cells in the columns that are spanned by this entry.

row

Indicates that the current entry is a header for all cells in the row.

rowgroup

Indicates that the current entry is a header for all cells in the rows that are spanned by this entry.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Universal attributes

This group defines a set of attributes that are available on almost all DITA elements. It includes all elements in the ID, localization, and metadata attribute groups, as well as the following attributes:

@class (not for use by authors)

This attribute is not for use by authors. If an editor displays @class attribute values, do not edit them. Specifies a default value that defines the specialization ancestry of the element. Its predefined values allow DITA tools to work correctly with specialized elements. In a generalized DITA document the @class attribute value in the generalized instance might differ from the default value for the @class attribute for the element as given in the DTD or schema. See [The class attribute rules and syntax](#) for more information. This attribute is specified on every element except for the <dita> container element. It is always specified with a default value, which varies for each element.

@outputclass

Specifies a role that the element is playing. The role must be consistent with the basic semantic and expectations for the element. In particular, the @outputclass attribute can be used for styling during output processing; HTML output will typically preserve @outputclass for CSS processing.

Comment by robander

I don't like "The role must be consistent...", that seems like best practice that cannot be normative – and I could easily say outputclass="flashy" which makes my element show up with sparkles, and has nothing to do with "the basic semantic and expectations for the element".

Disposition: Unassigned

B.2 Universal attribute group

The universal attribute group defines a set of common attributes that are available on almost every DITA element. The universal attribute group includes all attributes from the ID, localization, and metadata attribute groups, plus the @class and @outputclass attributes.

Comment by Kristen J Eberlein on 29 December 2021

This is something wrong with the organizational structure of this topic ... Look at it in outline form, and check that the sections, titles, and content all make logical sense with the topic title of "Universal attribute group".

Disposition: Unassigned

Common attribute groups

The following attribute groups are referenced in this specification. They are also used in the grammar files when the element attributes are defined.

Universal attributes

Includes @class and @outputclass, along with every attribute in the ID, localization, and metadata attribute groups.

ID attributes

This group includes the attributes that enable the naming and referencing of elements: @conaction, @conkeyref, @conref, @conrefend, and @id.

Localization attributes

This group includes attributes that are related to translation and localization: `@dir`, `@translate`, and `@xml:lang`.

Metadata attributes

Comment by Kristen J Eberlein on 31 December 2021

Why do we need to mention that two attributes are available for specialization here? I think it makes the paragraph hard to read.

Disposition: Unassigned

This group includes common metadata attributes, two of which are available for specialization: `@base`, `@importance`, `@props`, `@rev`, and `@status`.

The base DITA vocabulary from OASIS includes several specializations of `@props: @audience`, `@deliveryTarget`, `@otherprops`, `@platform`, and `@product`. These attributes are defined as attribute-extension domains. By default, they are integrated into all OASIS-provided document-type shells, but they can be made unavailable by implementing custom document-type shells.

Comment by Kristen J Eberlein on 29 December 2021

Why do we provide information about specialization and custom document-type shells here? I think that information could be removed.

Disposition: Unassigned

Universal attribute definitions

The universal attributes for OASIS DITA elements are defined below. Specialized attributes, which are part of the OASIS distribution but are only available when explicitly included in a shell, are noted in the list.

@audience (specialized attribute)

Indicates the intended audience for the element. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

@base

Specifies metadata about the element. It is often used as a base for specialized attributes that have a simple syntax for values, but which are not conditional processing attributes.

The `@base` attribute takes a space-delimited set of values. However, when serving as a container for generalized attributes, the attribute values will be more complex. See [Attribute generalization](#) for more details.

@class (not for use by authors)

This attribute is not for use by authors. If an editor displays @class attribute values, do not edit them. Specifies a default value that defines the specialization ancestry of the element. Its predefined values allow DITA tools to work correctly with specialized elements. In a generalized DITA document the `@class` attribute value in the generalized instance might differ from the default value for the `@class` attribute for the element as given in the DTD or schema. See [The class attribute rules and syntax](#) for more information. This attribute is specified on every element except for the `<dita>` container element. It is always specified with a default value, which varies for each element.

@conaction

Specifies how the element content will be pushed into a new location. The following values are valid:

mark

The element acts as a marker when pushing content before or after the target, to help ensure that the push action is valid. The element with `conaction="mark"` also specifies the target of the push action with `@conref`. Content inside of the element with `conaction="mark"` is not pushed to the new location.

pushafter

Content from this element is pushed after the location specified by `@conref` on the element with `conaction="mark"`. The element with `conaction="pushafter"` is the first sibling element after the element with `conaction="mark"`.

pushbefore

Content from this element is pushed before the location specified by `@conref` on the element with `conaction="mark"`. The element with `conaction="pushbefore"` is the first sibling element before the element with `conaction="mark"`.

pushreplace

Content from this element replaces any content from the element referenced by the `@conref` attribute. A second element with `conaction="mark"` is not used when using `conaction="pushreplace"`.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

See [B.4 STUB CONTENT](#) (68) for examples and details about the syntax.

@conkeyref

Specifies a key name or a key name with an element ID that acts as an indirect reference to reusable content. The referenced content is used in place of the content of the current element. See [B.4 STUB CONTENT](#) (68) for more details about the syntax and behaviors.

@conref

Specifies a URI that references a DITA element. The referenced content is used in place of the content of the current element. See [B.4 STUB CONTENT](#) (68) for examples and details about the syntax.

@conrefend

Specifies a URI that references the last element in a sequence of elements, with the first element of the sequence specified by `@conref`. The referenced sequence of elements is used in place of the content of the current element. See [B.4 STUB CONTENT](#) (68) for examples and details about the syntax.

@deliveryTarget (specialized attribute)

Specifies the intended delivery target of the content, for example, "html", "pdf", or "epub". If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

@dir

Identifies or overrides the text directionality. The following values are valid:

lro

Indicates an override of the Unicode Bidirectional Algorithm, forcing the element into left-to-right mode.

ltr

Indicates left-to-right.

rlo

Indicates an override of the Unicode Bidirectional Algorithm, forcing the element into right-to-left mode.

rtl

Indicates right-to-left.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

See [The dir attribute](#) for more information.

@id

Specifies an identifier for the current element. This ID is the target for references by `@href` and `@conref` attributes and for external applications that refer to DITA or LwDITA content. This attribute is defined with the XML data type NMTOKEN, except where noted for specific elements within the language reference.

See [id attribute](#) for more details.

@importance

Specifies the importance or priority that is assigned to an element. The following values are valid: "default", "deprecated", "high", "low", "normal", "obsolete", "optional", "recommended", "required", "urgent", and "-dita-use-conref-target". This attribute is not used for conditional processing, although applications might use the value of the `@importance` attribute to highlight elements. For example, in steps of a task topic, the value of the `@importance` attribute indicates whether a step is optional or required.

Comment by Kristen J Eberlein on 29 September 2022

I think the phrase "to highlight elements" is a little off. Maybe "render generated text"? And how about adding "Processors often add text or images to ensure that readers of the generated content understand whether the step is optional or required." to the end of the example?

Disposition: Unassigned

@otherprops (specialized attribute)

Specifies a property or properties that provide selection criteria for the element. Alternatively, the `@props` attribute can be specialized to provide a new metadata attribute instead of using the general `@otherprops` attribute. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

@outputclass

Specifies a role that the element is playing. The role must be consistent with the basic semantic and expectations for the element. In particular, the `@outputclass` attribute can be used for styling during output processing; HTML output will typically preserve `@outputclass` for CSS processing.

Comment by robander

I don't like "The role must be consistent...", that seems like best practice that cannot be normative – and I could easily say `outputclass="flashy"` which makes my element show up with sparkles, and has nothing to do with "the basic semantic and expectations for the element".

Disposition: Unassigned

@platform (specialized attribute)

Indicates operating system and hardware. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

Comment by robander

I think this could specify a platform that is not an operating system or hardware, right? The current definition explicitly limits platform to those two ... maybe "Specifies a platform or platforms to which the element applies, such as the operating system or hardware relevant to a task."

Disposition: Unassigned

@product (specialized attribute)

Specifies the name of the product to which the element applies. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

@props

Specifies metadata about the element. New attributes can be specialized from the @props attribute. This attribute supports conditional processing. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

The @props attribute takes a space-delimited set of values. However, when serving as a container for generalized attributes, the attribute values will be more complex. See [Attribute generalization](#) for more details.

@rev

Specifies a revision level of an element that identifies when the element was added or modified. It can be used to flag outputs when it matches a run-time parameter. It cannot be used for filtering nor is it sufficient to be used for version control. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

Comment by Kristen J Eberlein on 29 September 2022

I want to tweak this. How about the following? Also, neither definition describes what values are permitted.

Specifies metadata that identifies when the element was added or the content of the element was modified. The @rev attribute can be used for flagging. It cannot be used for filtering nor is it sufficient to be used for version control. If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

Disposition: Unassigned

@status

Specifies the modification status of the element. The following values are valid: "new", "changed", "deleted", "unchanged", and "-dita-use-conref-target".

@translate

Specifies whether the content of the element should be translated. The following values are valid: "yes", "no", and "-dita-use-conref-target".

See [Element-by-element recommendations for translators](#) for suggested processing defaults for each element.

Comment by Kristen J Eberlein on 31 December 2021

Does [Element-by-element recommendations for translators](#) really provide suggested processing defaults for each element? I thought it covered whether an element was block or in-line and whether there were considerations that translators needed to be aware of.

Disposition: Unassigned

@xml:lang

Specifies the language and optional locale of the content that is contained in an element. Valid values are language tokens or the null string. The @xml:lang attribute and its values are described in the [Extensible Markup Language 1.0 specification, fifth edition](#).

Comment by Kristen J Eberlein on 29 September 2022

Do we also want to direct readers to the architectural topics about the `@xml:lang` attribute?

Disposition: Unassigned

B.3 Common attributes

The common attributes topic collects defines most of the attributes that are used on more than one base element.

Common attribute groups

The following groups are referenced in this specification, and they are also used in grammar files when defining attributes for elements.

Architectural attributes

This group includes a set of attributes that are defined for document-level elements such as `<topic>` and `<map>`: `@DITAArchVersion`, `@specializations`, and `@xmlns:ditaarch`.

Common map attributes

This group includes attributes that are frequently used on map elements: `@cascade`, `@chunk`, `@collection-type`, `@keyscope`, `@linking`, `@processing-role`, `@search`, `@toc`, and `@subjectrefs`.

Complex table attributes

This group includes attributes that are defined on table elements but not simple table elements. These attributes are part of the OASIS Exchange Table Model, unless otherwise noted. Table elements generally use only a subset of the attributes that are defined in this group. This group contains the following attributes: `@align`, `@char`, `@charoff`, `@colsep`, `@rowheader`, `@rowsep`, and `@valign`.

Data-element attributes

Includes attributes defined on `<data>` and its many specializations: `@datatype`, `@name`, and `@value`

Date attributes

Includes attributes that take date values, and are defined on metadata elements that work with date information: `@expiry` and `@golive`

Display attributes

This group includes attributes that affect the rendering of many elements: `@exppanse`, `@frame`, and `@scale`.

Inclusion attributes

Includes attributes defined on `<include>` and its specializations: `@encoding` and `@parse`.

Link-relationship attributes

This group includes attributes whose values can be used for representing navigational relationships: `@format`, `@href`, `@type`, and `@scope`.

Simple table attributes

Comment by Kristen J Eberlein on 29 December 2021

If I have jumped to this place in a document from the element-reference topic, I want the attributes listed here in the "Simple table group" to be hyperlinked to the actual definition.

Disposition: Unassigned

This group includes attributes that are defined only on the `<simplatable>` element: `@keycol` and `@relcolwidth`. These attributes are listed in a group because the `<simplatable>` element is frequently used as a specialization base.

Table accessibility attributes

This group contains attributes that are defined on the `<stentry>` element and its specializations: `@headers` (61) and `@scope` (as defined on `<stentry>`) (66).

Other attributes (not in a group)

These are attributes that are used in the same way on more than one base element, but they are not formally grouped together: `@compact`, `@duplicates`, `@impose-role`, `@otherrole`, `@role`, and `@title-role`.

Common attribute definitions

Common attributes, including those in the groups listed above, are defined as follows.

@align (complex table attributes)

Specifies the **horizontal** alignment of text in table **entries**. The following values are valid:

left

Indicates left alignment of the text.

right

Indicates right alignment of the text.

center

Indicates center alignment of the text.

justify

Justifies the contents to both the left and the right.

char

Indicates character alignment. The text is aligned with the first occurrence of the character specified by the `@char` attribute.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

The `@align` attribute is available on the following table elements: `<colspec>`, `<entry>`, and `<tgroup>`.

@cascade (common map attributes)

Specifies how metadata attributes cascade within a map. The specification defines the following values:

merge

Indicates that the metadata attributes cascade, and that the values of the metadata attributes are additive. This is the processing default for the `@cascade` attribute.

nomerge

Indicates that the metadata attributes cascade, but that they are not additive for `<topicref>` elements that specify a different value for a specific metadata attribute. If the cascading value for an attribute is already merged based on multiple ancestor elements, that merged value continues to cascade until a new value is encountered. **That is**, setting `cascade="nomerge"` does not undo merging that took place on ancestor elements.

Processors can also define custom, implementation-specific tokens for this attribute.

See [Cascading of metadata attributes in a DITA map](#) for more information about how this attribute interacts with metadata attributes.

@char (complex table attributes)

Specifies the alignment character, which is the character that is used for aligning the text in table entries. This attribute applies when `align="char"`. A value of "" (the null string) means there is no aligning character.

For example, if `align="char"` and `char="."` are specified, then text in the table entry aligns with the first occurrence of the period within the entry. This might be useful if decimal alignment is required.

The `@char` attribute is available on the following table elements: `<colspec>` and `<entry>`.

@charoff (complex table attributes)

Specifies the horizontal offset of the alignment character **that is specified by the `@char` attribute**. The value is a greater-than-zero number that is less than or equal to 100. It represents the percentage of the current column width by which the text is offset to the left of the alignment character.

For example, if `align="char"`, `char="."`, and `charoff="50"` are all specified, then text in the table entry is aligned 50% of the distance to the left of the first occurrence of the period character within the table entry.

The `@charoff` attribute is available on the following table elements: `<colspec>` and `<entry>`.

@chunk (common map attributes)

Specifies how a processor should render a map or branch of a map. For example, it can be used to specify that individual topic documents should be rendered as a single document, or that a single document with multiple topics should be rendered as multiple documents.

The following values are valid:

combine

Instructs a processor to combine the referenced source documents for rendering purposes. **This** is intended for cases where a publishing process normally results in a single output artifact for each source XML document.

split

Instructs a processor to split each topic from the referenced source document into its own document for rendering purposes. **This** is intended for cases where a publishing process normally results in a single output artifact for each source XML document, regardless of how many DITA topics exist within each source document.

Processors can also define custom, implementation-specific tokens for this attribute.

For a detailed description of the `@chunk` attribute and its usage, see [Chunking](#).

@collection-type (common map attributes)

Specifies how topics or links relate to each other. The processing default is "unordered", although no default is specified in the OASIS-provided grammar files. The following values are valid:

unordered

Indicates that the order of the child topics is not significant.

sequence

Indicates that the order of the child topics is significant. Output processors will typically link between them in order.

choice

Indicates that one of the children should be selected.

family

Indicates a tight grouping in which each of the referenced topics not only relates to the current topic but also relate to each other.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@collection-type

The @collection-type attribute specifies how the children of a <topicref> element relate to their parent and to each other. This attribute, which is set on the parent element, typically is used by processors to determine how to generate navigation links in the rendered topics. For example, a @collection-type value of "sequence" indicates that children of the specifying <topicref> element represent an ordered sequence of topics; processors might add numbers to the list of child topics or generate next/previous links for online presentation. This attribute is available in topics on the <linklist> and <linkpool> elements, where it has the same behavior. Where the @collection-type attribute is available on elements that cannot directly contain elements, the behavior of the attribute is undefined.

Disposition: Unassigned

Comment by Kristen J Eberlein on 28 September 2022

In the definitions of the supported values, do we want to refer to "resources" instead of "topics"? Since we specify that @collection-type specifies "how topics **or** links relate to each other" ...

Disposition: Unassigned

@colsep (complex table attributes)

Specifies whether to render column separators between table entries. The following values are valid: "0" (no separators) and "1" (separators).

The @colsep attribute is available on the following table elements: <colspec>, <entry>, <table>, and <tgroup>.

@compact

Specifies whether the vertical spacing between list items is tightened. The following values are valid: "yes", "no", and "-dita-use-conref-target". Some DITA processors or output formats might not support the @compact attribute.

@datatype (data-element attributes)

Specifies the type of data contained in the @value attribute or within the <data> element. A typical use of @datatype will be the identifying URI for an XML Schema datatype.

@DITAArchVersion (architectural attributes)

Specifies the version of the DITA architecture that is in use. This attribute is in the namespace <http://dita.oasis-open.org/architecture/2005/>. This attribute is specified in the topic and map modules, and it uses a default value of the current version of DITA. The current default is "2.0".

@duplicates

Specifies whether duplicate links are removed from a group of links. Duplicate links are links that address the same resource using the same properties, such as link text and link role. How duplicate links are determined is processor-specific. The following values are valid:

yes

Specifies that duplicate links are retained.

no

Specifies that duplicate links are removed.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

The suggested processing default is "yes" within <linklist> elements and "no" for other links.

Comment by robander on Dec 28 2021

"How duplicate links are determined is processor-specific" ==> this should be included in any updates to standardize language around "implementation dependent".

Disposition: Unassigned

@encoding (inclusion attributes)

Comment by Kristen J Eberlein on 29 April 2019

Can we replace "should" in the following definition?

Disposition: Unassigned

Specifies the character encoding to use when translating the character data from the referenced content. The value should be a valid encoding name. If not specified, processors may make attempts to automatically determine the correct encoding, for example using HTTP headers, through analysis of the binary structure of the referenced data, or the <?xml?> processing instruction when including XML as text. The resource should be treated as UTF-8 if no other encoding information can be determined.

When `parse="xml"`, standard XML parsing rules apply for the detection of character encoding. The necessity and uses of `@encoding` for non-standard values of `@parse` are implementation-dependent.

@expansion (display attributes)

Specifies the horizontal placement of the element. The following values are valid:

column

Indicates that the element is aligned with the current column margin.

page

Indicates that the element is placed on the left page margin for left-to-right presentation or the right page margin for right-to-left presentation.

spread

Indicates that the object is rendered across a multi-page spread. If the output format does not have anything that corresponds to spreads, then "spread" has the same meaning as "page".

textline

Indicates that the element is aligned with the left (for left-to-right presentation) or right (for right-to-left presentation) margin of the current text line and takes indentation into account.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

For <table>, in place of the `@expansion` attribute that is used by other DITA elements, the `@pgwide` attribute is used in order to conform to the OASIS Exchange Table Model.

Some processors or output formats might not support all values.

@expiry (date attributes)

Specifies the date when the information should be retired or refreshed. The date is specified using the ISO 8601 format: `YYYY-MM-DD`, where `YYYY` is the year, `MM` is the month (01 to 12), and `DD` is the day (01-31).

@format (link-relationship attributes)

Specifies the format of the resource that is referenced. See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@frame (display attributes)

Specifies which portion of a border surrounds the element. The following values are valid:

all

Indicates that a line is rendered at the top, bottom, left, and right of the containing element.

bottom

Indicates that a line is rendered at the bottom of the containing element.

none

Indicates that no lines are rendered.

sides

Indicates that a line is rendered at the left and right of the containing element.

top

Indicates that a line is rendered at the top of the containing element.

topbot

Indicates that a line is rendered at the top and bottom of the containing element.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Some processors or output formats might not support all values.

@golive (date attributes)

Specifies the publication or general availability (GA) date. The date is specified using the ISO 8601 format: *YYYY-MM-DD*, where *YYYY* is the year, *MM* is the month (01 to 12), and *DD* is the day (01-31).

@headers

Specifies which entries in the current table provide headers for this cell. The `@headers` attribute contains an unordered set of unique, space-separated tokens, each of which is an ID reference of an entry from the same table.

@href (link-relationship attributes)

Specifies a reference to a resource. See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@impose-role

Specifies whether this element will impose its role on elements in a referenced map. The attribute is ignored if the target of the reference is not a map or branch of a map. The following values are valid:

keeptarget

The role of the current reference is not imposed on the target of the reference. This is the default for the unspecialized `<topicref>` element and for many convenience elements such as `<keydef>`.

impose

The role of the current reference is imposed on the target of the reference. For example, if a specialized topic reference `<chapter>` uses this value and references a map, a topic reference that resolves in place of the `<chapter>` will be treated as if it were a chapter.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@keycol (simpletable attributes)

Specifies the column that contains the content that represents the key to the tabular structure. If `@keycol` is present and assigned a numerical value, the specified column is treated as a vertical header.

@keyref

Specifies a key name that acts as a redirectable reference based on a key definition within a map. See [B.4 STUB CONTENT](#) (68) for information on using this attribute.

For HDITA, the equivalent of `@keyref` is `@data-keyref`

Comment by robander

The definition above for `@keyref` should be synchronized with the definition in the linked section on keys.

Disposition: Unassigned

@keys

Specifies one or more names for a resource. See [B.4 STUB CONTENT](#) (68) for information on using this attribute.

For HDITA, the equivalent of `@keys` is `@data-keys`

@keyscope (common map attributes)

Specifies that the element marks the boundaries of a key scope.

See [B.4 STUB CONTENT](#) (68) for information on using this attribute.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@keyscope

Defines a new scope for key definition and resolution, and gives the scope one or more names. For more information about key scopes, see [Indirect key-based addressing](#).

Disposition: Unassigned

@linking (common map attributes)

Specifies linking characteristics of a topic specific to the location of this reference in a map. If the value is not specified locally, the value might cascade from another element in the map (for cascade rules, see [Cascading of metadata attributes in a DITA map](#)).

Comment by robander on Dec 28 2021

The text below matches [1.3 spec text](#) but I'm nervous about "cannot link" type definition. It's describing how to generate links based on the current context in the map - it's not describing what the topic itself is allowed to link to, which is how I interpret "can".

Disposition: Unassigned

The following values are valid:

targetonly

A topic can only be linked to and cannot link to other topics.

sourceonly

A topic cannot be linked to but can link to other topics.

normal

A topic can be linked to and can link to other topics. Use this to override the linking value of a parent topic.

none

A topic cannot be linked to or link to other topics.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@linking

By default, the relationships between the topics that are referenced in a map are reciprocal:

- Child topics link to parent topics and vice versa.
- Next and previous topics in a sequence link to each other.
- Topics in a family link to their sibling topics.
- Topics referenced in the table cells of the same row in a relationship table link to each other. A topic referenced within a table cell does not (by default) link to other topics referenced in the same table cell.

This behavior can be modified by using the `@linking` attribute, which enables an author or information architect to specify how a topic participates in a relationship. The following values are valid:

linking="none"

Specifies that the topic does not exist in the map for the purposes of calculating links.

linking="sourceonly"

Specifies that the topic will link to its related topics but not vice versa.

linking="targetonly"

Specifies that the related topics will link to it but not vice versa.

linking="normal"

Default value. It specifies that linking will be reciprocal (the topic will link to related topics, and they will link back to it).

Authors also can create links directly in a topic by using the `<xref>` or `<link>` elements, but in most cases map-based linking is preferable, because links in topics create dependencies between topics that can hinder reuse.

Note that while the relationships between the topics that are referenced in a map are reciprocal, the relationships merely *imply* reciprocal links in generated output that includes links. The rendered navigation links are a function of the presentation style that is determined by the processor.

Disposition: Unassigned

@name (data-element attributes)

Defines a unique name for the object.

Comment by robander

Do we need to specify the scope of "unique" here?

Disposition: Unassigned

@otherrole

Specifies an alternate role for a link relationship when the `@role` attribute is set to "other".

@parse (inclusion attributes)

Specifies the processing expectations for the referenced resource. Processors must support the following values:

text

The contents should be treated as plain text. Reserved XML characters should be displayed, and not interpreted as XML markup.

xml

The contents of the referenced resource should be treated as an XML document, and the referenced element should be inserted at the location of the `<include>` element. If a fragment identifier is included in the address of the content, processors must select the element with the specified ID. If no fragment identifier is included, the root element of the referenced XML document is selected. Any grammar processing should be performed during resolution, such that default attribute values are explicitly populated. Prolog content must be discarded.

It is an error to use `parse="xml"` anywhere other than within `<foreign>` or a specialization thereof.

Processors may support other values for the `@parse` attribute with proprietary processing semantics. Processors should issue warnings and use `<fallback>` when they encounter unsupported `@parse` values. Non-standard `@parse` instructions should be expressed as URIs.

Note Proprietary `@parse` values will likely limit the portability and interoperability of DITA content, so should be used with care.

@processing-role (common map attributes)

Specifies whether the referenced resource is processed normally or treated as a resource that is only included in order to resolve references, such as key or content references. The following values are valid:

normal

Indicates that the resource is a readable part of the information set. It is included in navigation and search results. This is the default value for the `<topicref>` element.

resource-only

Indicates that the resource should be used only for processing purposes. It is not included in navigation or search results, nor is it rendered as a topic. This is the default value for the `<keydef>` element.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

If no value is specified but the attribute is specified on a containing element within a map or within the related-links section, the value cascades from the closest containing element.

@relcolwidth (simpletable attributes)

Specifies the width of each column in relationship to the width of the other columns. The value is a space-separated list of relative column widths. Each column width is specified as a positive integer or decimal number followed by an asterisk character.

For example, the value `relcolwidth="1* 2* 3*"` gives a total of 6 units across three columns. The relative widths are 1/6, 2/6, and 3/6 (16.7%, 33.3%, and 50%). Similarly, the value `relcolwidth="90* 150*"` causes relative widths of 90/240 and 150/240 (37.5% and 62.5%).

@role

Specifies the role that a linked topic plays in relationship with the current topic.

For example, in a parent/child relationship, the role would be "parent" when the target is the parent of the current topic, and "child" when the target is the child of the current topic. This can be used to sort and classify links when rendering.

The following values are valid:

ancestor

Indicates a link to a topic above the parent topic.

child

Indicates a link to a direct child such as a directly nested or dependent topic.

cousin

Indicates a link to another topic in the same hierarchy that is not a parent, child, sibling, next, or previous.

descendant

Indicates a link to a topic below a child topic.

friend

Indicates a link to a similar topic that is not necessarily part of the same hierarchy.

next

Indicates a link to the next topic in a sequence.

other

Indicates any other kind of relationship or role. The type of role is specified as the value for the `@otherrole` attribute.

parent

Indicates a link to a topic that is a parent of the current topic.

previous

Indicates a link to the previous topic in a sequence.

sibling

Indicates a link between two children of the same parent topic.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

@rowheader (complex table attributes)

Specifies whether the entries in the respective column are row headers. The following values are valid:

firstcol

Indicates that entries in the first column of the table are row headers. This applies when the `@rowheader` attribute is specified on the `<table>` element.

headers

Indicates that entries of the column that is described using the `<colspec>` element are row headers. This applies when the `@rowheader` attribute is specified on the `<colspec>` element.

norowheader

Indicates that entries in the first column are not row headers. This applies when the `@rowheader` attribute is specified on the `<table>` element.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

Note This attribute is not part of the OASIS Exchange Table Model upon which DITA tables are based. Some processors or output formats might not support all values.

The `@rowheader` attribute is available on the following table elements: `<table>` and `<colspec>`.

@rowsep (complex table attributes)

Specifies whether to render row separators between table entries. The following values are valid: "0" (no separators) and "1" (separators).

The @rowsep attribute is available on the following table elements: <colspec>, <entry>, <row>, <table>, and <tgroup>.

@scale (display attributes)

Specifies the percentage by which fonts are resized in relation to the normal text size. The value of this attribute is a positive integer. When used on <table> or <simpletable>, the following values are valid: "50", "60", "70", "80", "90", "100", "110", "120", "140", "160", "180", "200", and [-dita-use-conref-target](#) (68).

This attribute is primarily useful for print-oriented display. Some processors might not support all values.

If the @scale attribute is specified on an element that contains an image, the image is not scaled. The image is scaled **only** if a scaling property is explicitly specified for the <image> element.

@scope (link-relationship attributes)

Specifies the closeness of the relationship between the current document and the referenced resource. The following values are valid: "local", "peer", "external", and "-dita-use-conref-target".

See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@scope

Specifies that the current entry is a header for other table entries. The following values are valid:

col

Indicates that the current entry is a header for all cells in the column.

colgroup

Indicates that the current entry is a header for all cells in the columns that are spanned by this entry.

row

Indicates that the current entry is a header for all cells in the row.

rowgroup

Indicates that the current entry is a header for all cells in the rows that are spanned by this entry.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

@search (common map attributes)

Specifies whether the target is available for searching. If the value is not specified locally, the value might cascade from another element in the map (for cascade rules, see [Cascading of metadata attributes in a DITA map](#)). The following values are valid: "yes", "no", and "-dita-use-conref-target".

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@search

Specifies whether the topic is included in search indexes.

Disposition: Unassigned

@specializations (architectural attributes)

Specifies the attribute-domain specializations that are included in the document-type shell. This attribute is set as a default within the document-type shell. The value varies depending on what domains are integrated into the document-type shell. For example, a grammar file that includes the specialized attributes @audience, @deliveryTarget, and @newBaseAtt would set the value to @props/audience @props/deliveryTarget @base/newBaseAtt.

@subjectrefs (common map attributes)

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.

@title-role (REQUIRED)

Specifies the role that the alternative title serves. Multiple roles are separated by white space. The following roles are defined in the specification: "linking", "navigation", "search", "subtitle", and "hint".

Processors can define custom values for the `@title-role` attribute.

@toc (common map attributes)

Specifies whether a topic appears in the table of contents (TOC) based on the current map context. If the value is not specified locally, the value might cascade from another element in the map (for cascade rules, see [Cascading of metadata attributes in a DITA map](#)). The following values are valid:

yes

The topic appears in a generated TOC.

no

The topic does not appear in a generated TOC.

-dita-use-conref-target

See [B.4 STUB CONTENT](#) (68) for more information.

Comment by Kristen J Eberlein on 28 September 2022

Here is the content from the "DITA map attributes" topic:

@toc

Specifies whether topics are excluded from navigation output, such as a Web site map or an online table of contents. By default, `<topicref>` hierarchies are included in navigation output; relationship tables are excluded.

Disposition: Unassigned

@type (link-relationship attributes)

Describes the target of a reference. See [B.4 STUB CONTENT](#) (68) for detailed information on supported values and processing implications.

@value (data-element attributes)

Specifies a value associated with the current property or element.

@valign (complex table attributes)

Specifies the vertical alignment of text in table [entries](#). The following values are valid:

bottom

Indicates that text is aligned with the bottom of the table entry.

middle

Indicates that text is aligned with the middle of the table entry.

top

Indicates that text is aligned with the top of the table entry.

-dita-use-conref-target

See [Using the -dita-use-conref-target value](#) for more information.

The `@valign` attribute is available on the following table elements: `<entry>`, `<tbody>`, `<thead>`, and `<row>`.

@xml:space

Specifies how to handle white space in the current element. This attribute is provided on `<pre>`, `<line>`, and on elements specialized from those. It ensures that parsers respect white space that

is part of the data in those elements, including line-end characters. When defined, it has a fixed value of "preserve", making it a default property of the element that cannot be changed or deleted by authors.

@xmlns:ditaarch (architectural attributes)

Declares the default DITA namespace. This namespace is declared as such in the RNG modules for `<topic>` and `<map>`, but it is specified as an attribute in the equivalent DTD-based modules. The value is fixed to "http://dita.oasis-open.org/architecture/2005/".

B.4 STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

[STUB CONTENT](#) (69)

[STUB CONTENT](#) (69)

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

STUB CONTENT

Index

A

attribute groups

- architectural attributes [38](#)
- common map attributes [38](#)
- complex table attributes [38](#)
- data-element attributes [38](#)
- date attributes [38](#)
- display attributes [38](#)
- ID and conref attributes [38](#)
- inclusion attributes [38](#)
- link relationship attributes [38](#)
- localization attributes [38](#)
- metadata attributes [38](#)
- simple table attributes [38](#)
- universal [51](#)
- universal attributes [38](#)

C

- common attributes [56](#)

G

- `<glossAbbreviation>` [13](#)
- `<glossAcronym>` [14](#)
- `<glossAlt>` [16](#)
- `<glossAlternateFor>` [18](#)
- glossaries
 - abbreviations [13](#)
 - acronyms [14](#)
 - definitions [21](#)
 - entries [21](#)
 - extension points [25](#)
 - images [31](#)
 - parts of speech [23](#)
 - scope notes [26](#)
 - short forms of a term [28](#)
 - surface forms [29](#)
 - synonyms [32](#)
 - terms [33](#)
 - usage information [34](#)
 - usage status [28](#)
 - variant terms [16, 18](#)
- `<glossBody>` [20](#)
- `<glossdef>` [21](#)
- `<glossentry>` [21](#)
- glossentry elements [13](#)
 - `<glossAbbreviation>` [13](#)
 - `<glossAcronym>` [14](#)
 - `<glossAlt>` [16](#)
 - `<glossAlternateFor>` [18](#)
 - `<glossBody>` [20](#)
 - `<glossdef>` [21](#)

- `<glossentry>` [21](#)
- `<glossProperty>` [25](#)
- `<glossScopeNote>` [26](#)
- `<glossShortForm>` [28](#)
- `<glossStatus>` [28](#)
- `<glossSurfaceForm>` [29](#)
- `<glossSymbol>` [31](#)
- `<glossSynonym>` [32](#)
- `<glossterm>` [33](#)
- `<glossUsage>` [34](#)
- `<glossPartOfSpeech>` [23](#)
- `<glossProperty>` [25](#)
- `<glossScopeNote>` [26](#)
- `<glossShortForm>` [28](#)
- `<glossStatus>` [28](#)
- `<glossSurfaceForm>` [29](#)
- `<glossSymbol>` [31](#)
- `<glossSynonym>` [32](#)
- `<glossterm>` [33](#)
- `<glossUsage>` [34](#)

R

rendering expectations

- `<abbreviated-form>` [10](#)

U

- universal attribute group [51](#)