
Stage two: #351 Add multimedia elements to base

Add multimedia elements to base.

Date and version information

This proposal includes the following information:

Date that this feature proposal was completed

06 July 2020

Champion of the proposal

Kristen James Eberlein, Eberlein Consulting LLC

Links to any previous versions of the proposal

30 June 2020: [Request for early feedback](#)

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

The original proposal to add a multimedia domain was first discussed at the DITA TC meeting on [22 May 2017](#). Chris Nitchie, Oberon Technologies, then championed the proposal and carried it through stage two and stage three. On [24 March 2020](#), the DITA TC decided to revisit the proposal and add the multimedia elements to the base.

Reviewers for Stage 2 proposal

Carlos Evia, Virginia Tech

Chris Nitchie, Individual member

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

Not applicable

Link to the GitHub issue

[Issue 351: Add multimedia elements to base](#)

Original requirement or use case

DITA needs elements to explicitly reference audio and video content.

Originally, this was planned to be a DITA 1.3-compatible domain, designed to provide support for Lightweight DITA. Once it was clear that Lightweight DITA will be released at the same time as (or later than) DITA 2.0, the TC decided these elements should be base DITA 2.0 elements instead of a domain.

Use cases

DITA topics need to easily reference audio and video content.

New terminology

Not applicable.

Proposed solution

Four new, unspecialized elements will be added to the DITA 2.0 base:

<audio>

Description: Audio is sound that the human ear is capable of hearing.

Content model:

- Zero or one <desc>, <longdesc>, and <fallback>
- Zero or more <media-source>, <media-track>, and %foreign.unknown.incl;

Contained by: Everywhere that <object> is available

Attributes:

- Universal attributes
- @href, @keyref, and @type
- @autoplay, @controls, @loop, @muted, and @tabindex

<media-source>

Description: The media source specifies the location of an audio or video resource.

Content model: Empty

Contained by: <audio> and <video>

Attributes:

- Universal attributes
- @href, @keyref, and @type

<media-track>

Description: Media track settings specify the location of supplemental text-based data for the referenced media, for example, subtitles or descriptions.

Content model: Empty

Contained by: <audio> and <video>

Attributes:

- Universal attributes. Note that @xml:lang on <media-track> will specify the language of the target resource, analogous to @lang on <track> in HTML5.
- @href, @keyref, and @kind

<video>

Description: A video is a recording of moving visual images.

Content model:

- Zero or one <desc> and <fallback>
- Zero or more <media-source>
- Zero or more <media-track>

Contained by: Everywhere that <object> is available

Attributes:

- Universal attributes
- @href, @keyref, and @type
- @height and @width
- @autoplay, @controls, @loop, @muted, @poster, and @tabindex
- @posterkeyref

Benefits

This proposal addresses the following questions:

Who will benefit from this feature?

All DITA implementations with content that includes multimedia elements.

What is the expected benefit?

Ease of authoring DITA topics that reference multimedia content.

How many people probably will make use of this feature?

Many.

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

Significant.

Technical requirements

Adding new elements or attributes

See [Proposed solution](#) (1) above.

The `<fallback>` element already has been added to the DITA 2.0 base as part of the implemented proposal #27, "Multimedia domain."

Processing impact

Implementers will need to add the new markup to their processing logic for key resolution, especially `@posterkeyref` on the `<video>` element.

Tools and systems that manage, analyze, and report on relationships between objects will need to be updated to consider the `<audio>`, `<video>`, `<media-source>`, and `<media-track>` elements.

Most of the elements and attributes have one-to-one mappings to the equivalent HTML5 markup, thus making HTML conversions relatively simple. Conversions into other output formats, such as PDF, will require extra effort.

Overall usability

This proposal will make it much simpler to reference audio and video content from DITA topics.

Backwards compatibility

This proposal does not introduce any backwards incompatibility.

Migration plan

Not applicable.

Costs

This proposal will have an impact on the following groups:

Maintainers of the grammar files

The new elements will need to be added to the base.

Editors of the DITA specification

- Four new topics will be required. The topics introduced as part of issue #27, "Multimedia domain," can be reworked. These topics have already been through an extensive DITAweb review.
- Some existing topics will need to be edited: Container topic for body elements, recommendations for translators, etc.

Vendors of tools

Tool vendors will need to update the following:

- Authoring environments to simplify authoring of `<audio>` and `<video>` elements
- Processors to ensure proper handling of `<audio>` and `<video>` elements for various output formats

- Tools and systems that manage, analyze, and report on relationships between objects

DITA community-at-large

This proposal will make it much easier for DITA authors to code references to audio and video content, thus reducing the perception of DITA complexity.

Producing migration instructions or tools

Not applicable

Examples

This section contains example of the markup for the new elements.

Figure 1: An <audio> element with a single source

```
<audio href="mysong.mp3"/>
```

Figure 2: An <audio> element with three formats

```
<audio href="mysong.mp3" type="audio/mpeg">  
  <fallback>Audio playback is unavailable.</fallback>  
  <media-source href="mysong.wav" type="audio/wav" />  
  <media-source href="mysong.ogg" type="audio/ogg" />  
</audio>
```

Figure 3: An <audio> element with controls turned on and looping turned off

```
<audio href="mysong.mp3" autoplay="false" controls="true" loop="false">  
  <fallback>Audio playback is unavailable</fallback>  
</audio>
```

Figure 4: A <video> element referenced by key

```
<video keyref="exampleVideo"/>
```

Figure 5: A fuller example of a <video> element

```
<video width="500px" height="300px"  
  autoplay="true" controls="false" poster="poster.png">  
  <fallback>Video playback not available.</fallback>  
  <media-source href="myvideo.mp4" kind="video/mpeg-4"/>  
  <media-source href="myvideo.ogg" kind="video/ogg"/>  
  <media-track href="myvideo-subtitles-en.vtt" kind="subtitles" xml:lang="en"/>  
  <media-track href="myvideo-subtitles-fr.vtt" kind="subtitles" xml:lang="fr"/>  
</video>
```