This is a Non-Standards Track Work Product. The patent provisions of the OASIS IPR Policy do not apply.

Lightweight DITA: An Introduction

Working Draft 09

08 March 2017

Specification URIs

This version:
http://docs.oasis-open.org/dita/LwDITA/v1.0/cn01/LwDITA-v1.0-cn01.html
(Authoritative)
http://docs.oasis-open.org/dita/LwDITA/v1.0/cn01/LwDITA-v1.0-cn01.pdf

Previous version:
Not applicable.

Latest version:
http://docs.oasis-open.org/dita/LwDITA/v1.0/LwDITA-v1.0.html
(Authoritative)
http://docs.oasis-open.org/dita/LwDITA/v1.0/LwDITA-v1.0.pdf

Technical Committee:
OASIS Darwin Information Typing Architecture (DITA) TC

Chair:
Kristen James Eberlein (kris@eberleinconsulting.com), Eberlein Consulting LLC

Editors:
Carlos Evia (cevia@vt.edu), Virginia Tech
Kristen James Eberlein (kris@eberleinconsulting.com), Eberlein Consulting LLC

Additional artifacts:
This document is part of a work product that also includes:

- ZIP file that contains the DITA source for this document. http://docs.oasis-open.org/dita/LwDITA/v1.0/cn01/LwDITA-v1.0-cn01-DITA-source.zip

Related work:
This document is related to:

- Darwin Information Typing Architecture (DITA) Part 1: Base Edition. http://docs.oasis-open.org/dita/dita/v1.3/dita-v1.3-part1-base.html. This edition contains topic and map; it is designed for implementers and users who need only the most fundamental pieces of the DITA framework.


Abstract:
Lightweight DITA (LwDITA) is a slimmed-down version of DITA that is designed to ease adoption and implementation of DITA.

LwDITA provides a subset of the functionality of the full DITA standard. It has a limited element and attribute set, stricter content models, support for non-XML languages such as HTML5 and Markdown.

Status:
This document was last revised or approved by the OASIS Darwin Information Typing Architecture (DITA) TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document.

TC members should send comments on this document to the TC’s email list. Others should send comments to the TC’s public comment list, after subscribing to it by following the instructions at the “Send A Comment” button on the TC’s web page at https://www.oasis-open.org/committees/comments/index.php?wg_abbrev=dita.

Citation format:
When referencing this note, the following citation format should be used:

[LwDITA-intro-v1.0]

Notices
Copyright © OASIS Open 2017. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules
Table of contents

1 Introduction.................................................................................................................................. 5
  1.1 References............................................................................................................................ 5
  1.2 Terminology.......................................................................................................................... 6
2 Why Lightweight DITA?................................................................................................................ 7
  2.1 Simplified structure.............................................................................................................. 7
  2.2 Support for non-XML formats..............................................................................................8
3 Lightweight DITA design............................................................................................................... 9
  3.1 Subset of DITA 1.3 elements................................................................................................9
  3.2 Stricter content model....................................................................................................... 10
  3.3 Subset of reuse mechanisms............................................................................................. 11
  3.4 New multimedia elements................................................................................................. 12
  3.5 New elements for footnotes.............................................................................................. 13
  3.6 LwDITA document types................................................................................................14
    3.6.1 LwDITA topic............................................................................................................... 14
    3.6.2 LwDITA map................................................................................................................ 14
4 LwDITA authoring formats..........................................................................................................15
  4.1 XDITA.................................................................................................................................. 15
    4.1.1 Audience for XDITA.................................................................................................... 15
    4.1.2 Example of an XDITA topic......................................................................................... 16
    4.1.3 Example of an XDITA map......................................................................................... 17
  4.2 HDITA.................................................................................................................................. 17
    4.2.1 Audience for HDITA....................................................................................................17
    4.2.2 Example of an HDITA topic........................................................................................ 18
    4.2.3 Example of an HDITA map......................................................................................... 18
  4.3 MDITA................................................................................................................................. 19
    4.3.1 Audience for MDITA................................................................................................... 19
    4.3.2 Examples of MDITA topics......................................................................................... 19
    4.3.3 Example of an MDITA map........................................................................................ 21
  4.4 Authoring cross-format content with LwDITA....................................................................21
    4.4.1 Cross-format example: DITA map............................................................................... 22
    4.4.2 Cross-format example: XDITA topic........................................................................... 22
    4.4.3 Cross-format example: HDITA topic.......................................................................... 23
    4.4.4 Cross-format example: MDITA topic.......................................................................... 23
5 Lightweight DITA tools................................................................................................................25
Appendix A LwDITA elements and attributes................................................................................26
  Appendix A.1 DITA 1.3 elements in LwDITA.............................................................................26
  Appendix A.2 New elements.................................................................................................... 29
  Appendix A.3 DITA 1.3 attributes in LwDITA............................................................................29
Appendix B Acknowledgments......................................................................................................31
Appendix C Revision history..........................................................................................................32
1 Introduction

Lightweight DITA (LwDITA) is a slimmed-down version of the Darwin Information Typing Architecture (DITA). It is designed to ease adoption and implementation of DITA.

This committee note explains the rationale for LwDITA. It also provides an overview of the audiences for LwDITA and information about its three authoring formats.

1.1 References

The following are references to external documents or resources that readers of this document might find useful.

[CommonMark]

[GFM]
Writing on GitHub. https://help.github.com/categories/writing-on-github/.

[HTML5]

[LwDITA-cross-format-content]

[LwDITA-pre/overview]

[LwDITA]

[LwDITA-IXIASOFT]

[Markingdown-DITA]
1.2 Terminology

This section provides information about terminology and how it is used in this committee note.

Comment by Carlos Evia on 8 March 2017
This topic needs to be updated

ATX headers

(MDITA) One or two hash (#) marks at the beginning of a line of text. One hash mark indicates a topic title, and two hash marks indicates a section title.

document type

A type of DITA topic or map that is designed for a specific purpose.
2 Why Lightweight DITA?

Lightweight DITA (LwDITA) is a slimmed-down version of DITA that is designed to ease adoption and implementation of DITA. It also adds support for authoring in HTML5 and Markdown.

The DITA standard's core features are semantic tagging, chunking of content into topics, content re-use at various levels, and options for conditional publishing. LwDITA offers all of these features. However, whereas full DITA also contains specialized content types for technical content and learning and training, advanced elements and attributes customizable for a broad range of use-cases, LwDITA deliberately limits itself to generically usable constructs. This results in a much smaller standard in terms of elements, attributes, and complexity.

This pragmatic design has implications for small projects and large content collections alike. For individuals, learning how to use the standard in a sensible way will be much easier. LwDITA is designed to allow the use of more generic authoring workflows than full DITA. For organizations, implementing LwDITA will involve much less change-management effort and lower costs than those associated with full DITA.

The choices that keep LwDITA lightweight also mean that this proposed standard has functional limitations compared to full DITA. For users who -after having gained experience with LwDITA in structured content authoring- need more advanced features for customization and diverse content domains, LwDITA serves as a good start to evolve into full DITA.

2.1 Simplified structure

Full DITA has more power (and thus complexity) than is needed in some situations. LwDITA provides a slimmed-down alternative.

Conference presentations and practitioners' blogs occasionally describe DITA as an intimidating language with too many element types. In the all-inclusive edition, the DITA 1.3 standard has 26 document types and 621 element types. Even in the base edition, DITA 1.3 has four document types and 189 element types. In contrast, LwDITA has one topic type that can contain a maximum of 41 element types.

Although the many features and strengths of full DITA are undoubtedly useful for large content collections and environments that require multiple deliverables in many platforms for diverse audiences, this comes with the added cost of complexity. LwDITA seeks to address this complexity, and it helps overcome some of the barriers to adoption that are faced by full DITA.

LwDITA's core design principle is that a simplified DITA will be an easier entry point to DITA for both authors and vendors building commercial tools and applications.
2.2 Support for non-XML formats

LwDITA adds support for structured authoring in HTML5 and Markdown.

New forms of non-XML structured authoring have gained popularity. Authors are using the extended semantic markup of HTML5 to create structured documents for the Web. Many in industry and academia have also adopted plain text languages like Markdown.

In its initial release, LwDITA has three authoring formats:

- XDITA: A XML-based variant
- HDITA: An HTML5-based variant
- MDITA: A Markdown-based variant

Comment by Carlos Evia on 5 January 2017
Add para here mentioning that LwDITA can expand to include additional authoring formats?

These three authoring formats do not represent a final version of LwDITA. The Lightweight DITA Subcommittee decided to focus on XML, HTML5, and Markdown for this initial release based on members’ interest and expertise. LwDITA can expand to additional authoring formats (e.g., JSON, AsciiDoc, MS Word) based on audience interest and design capability in the Subcommittee membership.

HDITA and XDITA are designed to be fully compatible with each other, while MDITA is a compatible subset. XDITA and HDITA conform with the OASIS DITA and W3C HTML5 standards, respectively. In its core profile, MDITA aligns with the CommonMark effort for a Markdown standard, while also adopting some elements of GitHub Flavored Markdown and YAML. In its extended profile, MDITA can incorporate XDITA elements and attributes to overcome Markdown’s limitations as a language for authoring structured and reusable content.

Comment by Kristen J Eberlein on 30 November 2016
This will do for now, but I think we are going to need to be very rigorous about how we talk about compatibility and conformance. I expect that we will need a separate topic (topics?) to cover this.

The multiple authoring formats are designed to enable authors to continue working with the type of structured content – XML, HTML, or Markdown – that they are comfortable with and currently using. Rather than asking authors to adopt a single authoring platform, LwDITA provides mappings from one authoring format to another.

LwDITA also acts an entry point to full DITA for those who need more functionality and richer semantic structures.
3 Lightweight DITA design

LwDITA is designed to have a smaller element set, a stricter content model, and fewer reuse mechanisms than DITA 1.3. However, LwDITA also includes new elements and attributes that are added to provide increased multimedia support.

3.1 Subset of DITA 1.3 elements

LwDITA is designed to use a strict subset of the elements that are available in DITA 1.3. The subset was carefully chosen to include the most basic constructions that are needed in order to structure information effectively. Diverse industries and sectors – education, engineering, healthcare, and marketing – were considered.

The elements selected represent the following types of information:

- Alternate text
- Body
- Cross reference
- Data
- Description
- Figure
- Footnote
- Image
- In-line formatting: Bold, italics, underline, super script, subscript
- Lists
  - Definition list
  - List item
  - Ordered list
  - Unordered list
- Map
- Note
- Paragraph
- Phrase
- Prolog
- Preformatted text
- Section
- Short description
- Table
- Title
- Topic

The intersection of LwDITA and DITA 1.3 includes 37 elements.
Need to revise this number when we are done with mappings.

For a complete list of the DITA 1.3 elements that are included in LwDITA, and how they are represented in LwDITA's authoring formats, see DITA 1.3 elements in LwDITA (26).

### 3.2 Stricter content model

LwDITA has a much stricter content model than DITA 1.3. This ensures a predictable markup structure in topics that simplifies reuse, transformations, style sheet logic, and tools development.

LwDITA's stricter content model (compared to those included in DITA 1.3) minimizes authoring decisions by presenting limited choices for elements and attributes. These minimized model, however, depends on a few strict structuring rules. For example, in XDITA and HDITA (the LwDITA authoring formats based on XML and HTML5, respectively), all text must be within paragraph (`<p>`) elements. Exceptions are the description, short description, and title elements. Within paragraphs, the following inline elements can appear:

- Bold
- Italics
- Phrase
- Superscript
- Subscript
- Underline (only available in XDITA)

*Comment by Kristen J Eberlein on 09 January 2017*

Feedback from Ullakaisa Kalanfer and others at Citec:

Other exceptions are `<title>` and `<desc>` elements.

They agree with Deb Bissantz that it would be handy to have a list of block elements.

In DITA 1.3, the following markup is valid:

```xml
<section>Compatible light bulbs include the following:
  <ul>
    <li>Compact Fluorescent</li>
    <li>Light Emitting Diode</li>
  </ul>
</section>
```

In contrast, in XDITA, the following markup must be used:

```xml
<section>
  <p>Compatible light bulbs include the following:</p>
  <ul>
    <li>
      <p>Compact Fluorescent</p>
    </li>
  </ul>
</section>
```
3.3 Subset of reuse mechanisms

LwDITA has a smaller set of reuse mechanisms than DITA 1.3. Due to syntax limitations, only a few of these reuse mechanisms work across all three LwDITA authoring formats.

**Conditional processing**

The only conditional processing attribute is the `@props` attribute.

**Content reference**

The `@conref` attribute is available only on the following elements:

- List item
- Ordered list
- Paragraph
- Section
- Table
- Unordered list

---

**Comment by carloseviapuerto on 23 January 2017**

Should this list be more "list item, ordered list, paragraph... " instead of the raw XML element? After all, we are also covering their equivalents in MDITA and HDITA.

---

**Comment by Kristen J Eberlein on 09 January 2017**

Feedback from Ullakaisa Kalanfer and others at Citec:

Why isn't `@conref` available on `<dl>` and `<note>`?

That also begs the question of `<dlentry>`, `<dt>`, and `<dd>` ...

---

**Key reference**

The `@keyref` attribute is available only on the Phrase or Span element.

**Variable text**

For variable text, such as product names, authors can use `@keyref` on Phrase or Span.

---

**Comment by Kristen J Eberlein on 2 December 2016**

I think we should add some of the reasons for these design choices.

---

**Comment by Carlos Evia on 15 December 2016**

Note that all text is wrapped in `<p>` elements.

In MDITA (the LwDITA authoring format based on Markdown), the minimized content model benefits from Markdown's simple information structures for paragraphs, lists, and other block elements.
Need to include content here to address reuse in HDITA and MDITA.

**Comment by Carlos Evia on 16 December 2016**

This topic looks very restrictive: authors are limited to and can't... Instead, should focus on what LwDITA allows by keeping things simple. Consider non-DITA audiences..

**Comment by Kristen J Eberlein on 21 December 2016**

I think we need to accept the fact that our primary audience – at this point – is the DITA TC and current users of DITA. They need to know in clear terms exactly what the design for XDITA is.

Yes, HDITA and MDITA are important, but I think trying to amalgamate everything together – or add disclaimers as you did in the "Stricter content model" are getting in the way of our ability to articulate the initial design.

If DITA is the parent, then XDITA is the child. I think HDITA is a cousin that has been adopted – and XDITA is trying to catch up to her facility with multimedia. MDITA is the youngest adopted child and maybe has some developmental limitations, although some people think he is most fun to play with.

This design simplifies the DITA authoring experience, as there are no choices to be made. To reuse block-level content, authors will use `<conref>`. For phrase-level content, authors will use `<keyref>`.

For a complete list of the DITA 1.3 attributes that are included in LwDITA, and how they are represented in LwDITA’s authoring formats, see [DITA 1.3 attributes in LwDITA](#).

### 3.4 New multimedia elements

LwDITA adds new elements for multimedia content. These elements are designed for compatibility with HTML5.

For years, authors have found different approaches to embed multimedia content in DITA-based deliverables for web distribution. The DITA 1.3 specification recommends the `<object>` element to include multimedia content in a topic, pointing out that it corresponds to the `<object>` element in HTML. However, one of HTML5’s key features was the introduction of direct elements for audio and video, which made some instances of the `<object>` element usage obsolete. LwDITA updates the XML-to-HTML element correspondence and introduces the following multimedia elements, which are specialized from the DITA 1.3 `<object>` element:

**Audio**

**Comment by carloseviapuerto**

Same here: should this list be more the name of the element instead of its XML representation? We are also covering HDITA and MDITA in this topic.

Audio is a link to sound to be included in the content

**Controls**

**Comment by carloseviapuerto on 17 January, 2017**
Apparently at one point controls was going to accept y/n as a value... but that changed. What kind of content goes in it? Question for Mark G. or Michael P.

Controls enable user interfaces for video playback and volume in Web-aimed transformations

**Fallback**

Fallback is content, or a link, presented as alternative to media resources of audio or video.

**Poster**

Poster is a link to an image or static video frame

**Source**

Source is a link to media resources of audio or video content

**Track**

Track is a link time-based text data relevant to audio or video content

**Video**

Video is a link to an audiovisual product to be included in the content

### 3.5 New elements for footnotes

LwDITA introduces an enhanced structure for footnotes, including support for cross referenced content.

LwDITA includes a modified element for footnote, based on the DITA 1.3 `<fn>`. In XDITA, `<fn>` is a block element, available in all block contexts, with a required id to force the use-by-reference model included in the DITA 1.3 specification. XDITA adds the element `<fnref>` as a new inline element, specialized from `<xref>`, with a fixed type value of `@fn`. The following elements add footnote support to LwDITA:

**Footnote**

Footnote indicates the source for facts or other material used in a text.

**Footnote cross reference**

Footnote cross reference provides a link to a different location within the current topic, a different topic, a specific location in another topic, or an external resource.

A LwDITA footnote can be represented, with modifications to accommodate different authoring languages, in XDITA, HDITA, and MDITA.

For a complete list of the DITA 1.3 elements that are included in LwDITA, and how they are represented in LwDITA’s authoring formats, see *DITA 1.3 elements in LwDITA* (26).
3.6 LwDITA document types

LwDITA includes document types for topic and map.

3.6.1 LwDITA topic

The LwDITA topic contains the fundamental structures that are necessary to author a unit of stand-alone information.

A LwDITA topic can contain the following structural elements:

**Title**
- A label that connotes the purpose of the content that is associated with it.

**Short description**
- A brief depiction of the purpose or theme of a topic.

**Prolog**
- A container for metadata about a topic (for example, author information or subject category).

**Body**
- A container for the main content of a topic. It might include several sections.

**Section**
- An organizational division within a topic. It can have an optional title.

A LwDITA topic can be represented, with modifications to accommodate different authoring languages, in XDITA, HDITA, and MDITA.

3.6.2 LwDITA map

The LwDITA map contains the fundamental structures that are necessary to create a content collection model and establish relationships among resources.

A LwDITA map can contain the following structural elements:

**Data**

**Map**
- A map describes the relationships among a set of resources, such as LwDITA and DITA topics

**Navigation title**

**Topic metadata**

**Topic reference**

A LwDITA map can be represented, with modifications to accommodate different authoring languages, in XDITA, HDITA, and MDITA.
4 LwDITA authoring formats

LwDITA simplifies the DITA experience by offering authoring formats based on a subset of DITA 1.3 XML, HTML5, and Markdown. These formats can work independently but can also support the creation of cross-format content.

Comment by Carlos Evia on 19 January 2017
Do we need more here than what we already have in the "Support for non-XML formats" topic? "HDITA and XDITA are designed to be fully compatible with each other, while MDITA is a compatible subset that depends on some HDITA elements and attributes to overcome Markdown's limitations as a language for authoring structured and reusable content. XDITA and HDITA conform with the OASIS DITA and W3C HTML5 standards, respectively. MDITA aligns with the CommonMark effort for a Markdown standard, while also adopting some elements of GitHub Flavored Markdown and YAML."

4.1 XDITA

XDITA is the authoring format of LwDITA that uses XML to structure information. XDITA is a subset of DITA, with new multimedia elements added to support interoperability with HTML5.

Comment by Carlos Evia on 5 December 2016
After we introduce the three LwDITA "flavors," we will need a topic to show how they play together: Maps in XDITA, MDITA, and HDITA referencing X, H, and M topics.

4.1.1 Audience for XDITA

XDITA is designed to be used by individuals who want to author DITA content but whom do not want (or need) the full power of DITA.

Example of potential users of XDITA include the following:

- Information developers who use an XML editor but who want a smaller set of elements and attributes with which to work
- Departments who want to reduce the cost of developing and maintaining style sheets by eliminating mixed content

Comment by carloseviapuerto
John Hunt says: "Can this be recast more generally, as processing or rendering?" Not sure how to address that...

Comment by carloseviapuerto on 3 February 2017
Ullakaisa says "But isn't the normal LW-DITA standard full of mixed content? Do you mean to say "reducing the set of elements", which it effectively does.

- Content developers who want their DITA content to be subsumed by a product documentation set that is based on Markdown or HTML5
4.1.2 Example of an XDITA topic

The following topic is authored in XDITA. In addition to basic DITA elements, note the new `<video>` element that is highlighted in bold.

Comment by Kristen J Eberlein on 24 November 2016

Is the markup for the `<data>` element what we want to recommend? Would the following be preferable?

```xml
<data name="author">
  <data value="Kevin Lewis"/>
</data>
```

Or the following?

```xml
<data name="author" value="Kevin Lewis"/>
```

Comment by Carlos Evia on 30 November 2016

K. Eberlein's recommendation for data is correct and I have updated the example.

```xml
<topic id="install-and-setup">
  <title>Installing and Setting up Remote Lighting</title>
  <shortdesc>Installation of your lighting kit includes installing the light bulbs into light fixtures, preparing the remote control, and programming lighting groups.</shortdesc>
  <prolog>
    <data name="author" value="Kevin Lewis"/>
  </prolog>
  <body>
    <section>
      <title>Steps</title>
      <ul>
        <li><p>Install light bulbs.</p></li>
        <li><p>Prepare remote control.</p></li>
        <li><p>Program lighting groups.</p></li>
      </ul>
    </section>
    <section>
      <title>Example</title>
      <p>The following video demonstrates a recommended installation:</p>
      <video>
        <controls />
        <source value="remote.mp4" />
      </video>
    </section>
  </body>
</topic>
```
XDITA topics are designed to be fully compatible with DITA topics. An author can work on an XDITA topic and keep it in a collection of LwDITA topics, but that same topic will also be compatible with maps and topics authored in DITA 1.3.

4.1.3 Example of an XDITA map

The following map is authored in XDITA.

```xml
<map id="remote-main">
  <topicmeta>
    <navtitle>Remote Lighting Network</navtitle>
  </topicmeta>
  <topicref href="introduction.dita">
    <topicmeta>
      <navtitle>Introduction</navtitle>
    </topicmeta>
  </topicref>
  <topicref href="alternatives.dita">
    <topicmeta>
      <navtitle>Alternative lighting setups</navtitle>
    </topicmeta>
  </topicref>
  <topicref href="low-power.dita">
    <topicmeta>
      <navtitle>Low power installation</navtitle>
    </topicmeta>
  </topicref>
  <topicref href="high-power.dita">
    <topicmeta>
      <navtitle>High power installation</navtitle>
    </topicmeta>
  </topicref>
</map>
```

Comment by Carlos Evia on 5 December 2016

Next example will show how to use keys and other attributes in XDITA

4.2 HDITA

HDITA is the authoring format of LwDITA that uses HTML5 to structure information. It also uses custom data attributes to provide interoperability with DITA.

4.2.1 Audience for HDITA

HDITA is designed to be used by individuals who want to author structured content using tools that are designed for HTML authoring.

Potential users of HDITA might include the following:

- Marketing writers who want to contribute to DITA-based product documentation without using an XML editor
• Software developers who want to contribute to documentation using tools for authoring HTML content
• Teachers and trainers who want to create course content for a Web site or learning management system (LMS)
• Bloggers and content strategists who want to be able to create and edit content using mobile devices

4.2.2 Example of an HDITA topic
An HDITA topic is authored in HTML5. It also includes custom data attributes to enable interoperability with DITA.

```html
<meta name="author" content="Kevin Lewis">
<article id="install-and-setup">
  <h1>Installing and Setting up Remote Lighting</h1>
  <p data-hd-class="shortdesc">Installation of your lighting kit includes installing the light bulbs into light fixtures, preparing the remote control, and programming lighting groups.</p>
  <h2>Steps</h2>
  <ul>
    <li><p>Install light bulbs.</p></li>
    <li><p>Prepare remote control.</p></li>
    <li><p>Program lighting groups.</p></li>
  </ul>
  <h2>Example</h2>
  <p>The following video demonstrates a recommended installation:</p>
  <video src="remote.mp4" controls poster="remote.png" />
</article>
```

HDITA topics are designed to be fully compatible with XDITA and DITA topics. Additionally, because HDITA is based on the HTML5 standard, an author can "round-trip" an HDITA topic and keep working on it as an independent HTML5 file. The HDITA custom data attributes for content reuse and compatibility with DITA will not affect validation of an HDITA topic as HTML5.

4.2.3 Example of an HDITA map
An HDITA map is authored in HTML5

```html
<nav>
  <h1>Remote Lighting Network</h1>
  <ul>
    <li><a href="introduction.html">Introduction</a></li>
    <li><a href="alternatives.html">Alternative lighting setups</a></li>
    <li><a href="low-power.html">Low power installation</a></li>
    <li><a href="high-power.html">High power installation</a></li>
  </ul>
</nav>
```
4.3 MDITA

MDITA is the authoring format of LwDITA that uses Markdown to structure information.

4.3.1 Audience for MDITA

MDITA is designed to be used by individuals who want to author structured content with the minimum of overhead but at the same time want to take advantage of reuse and publishing benefits associated with the DITA standard.

Example of potential users of MDITA include the following:

- Software developers who want to contribute to DITA-based product documentation without using an XML editor
- Software developers who want to contribute to product documentation using the tools and markup of their choice
- Individuals authoring content using a platform, such as a mobile device, that does not support an XML editor
- Individuals authoring content quickly that must be later refactored as structured content

4.3.2 Examples of MDITA topics

An MDITA topic is authored in Markdown. MDITA topics can be created following core or extended profiles.

**MDITA Core Profile**

The MDITA core profile contains simple, straight-forward information structures that are readily available in Markdown:

- Title
- Paragraph
- Section title
- Section
- Unordered list

The MDITA core profile aligns with the CommonMark proposal for a Markdown standard. The following example shows an MDITA core profile topic:

```
# Installing and Setting up Remote Lighting

Installation of your lighting kit includes installing the light bulbs into light fixtures, preparing the remote control, and programming lighting groups.

## Steps
```
1. Install light bulbs.
2. Prepare remote control.
3. Program lighting groups.

## Example

A video could demonstrate a recommended installation.

A topic created in the MDITA core profile does not have a short description, which can be provided using a `<topicmeta>` element in a DITA map. The required topic @id attribute is generated following a scheme that 1) takes the topic's title, 2) replaces blank spaces with underscores, and 3) replaces characters that are not allowed in URLs with a period (.) and the required character code.

### MDITA Extended Profile

The MDITA extended profile acknowledges the limitations of Markdown as a language for structuring content. This profile allows the following elements to enhance interoperability with other LwDITA authoring formats and full DITA:

- An optional YAML front matter header. This YAML header can supply a direct value for the @id attribute that is required on the root element of a DITA topic; it can also provide a short description and include prolog metadata about who authored the DITA topic. If included in a topic, the YAML front matter header must be the first thing in the MDITA file and must be set between triple-dashed lines.

Comment by Carlos Evia on 20 February 2017

The subcommittee needs to evaluate the use of YAML in MDITA during our initial web review

- Optional raw XDITA or HDITA attributes and elements. Although MDITA allows for this kind of syntax extension, its validation will depend on specific implementations. The following example shows an MDITA extended profile topic with a YAML header indicating its @id and short description, and an XDITA element that enables the topic to reference a video (indicated in bold text)

```---
id: install-and-setup
shortdesc: Installation of your lighting kit includes installing the light bulbs into light fixtures, preparing the remote control, and programming lighting groups.
author: Kevin Lewis
---

# Installing and Setting up Remote Lighting

Before you attempt to install your lighting kit, please turn off the power in your electrical circuit panel,

## Steps

1. Install light bulbs.
2. Prepare remote control.
3. Program lighting groups.
```
## Example

The following video demonstrates a recommended installation:

```xml
<video>
  <controls />
  <source value="remote.mp4" />
</video>
```

MDITA topics are designed as a compatible subset of XDITA and HDITA topics. Because of Markdown's limitations as a language for structured content, an MDITA extended profile topic will work in a LwDITA collection; however, that same topic might not "round-trip" to other Markdown-only environments.

### 4.3.3 Example of an MDITA map

An MDITA map is authored in Markdown.

```
# Remote Lighting Network

- [Introduction](introduction.md)
- [Alternative lighting setups](alternatives.md)
  - [Low power installation](low-power.md)
  - [High power installation](high-power.md)
```

Comment by Carlos Evia on 5 December 2016

Next example will show how to use keys and other attributes in MDITA

### 4.4 Authoring cross-format content with LwDITA

LwDITA enables cross-format content sharing. Authors can create topics in XDITA, HDITA, or MDITA and then combine them in a collection with a sub-set of DITA's reuse capabilities.

In the following example, a hypothetical team developing content for a lighting product's manual shares topics authored in LwDITA's different formats. The team even takes advantage of DITA reuse mechanisms like `@conref` and `@keyref`. The example contains the following:

- A DITA map that links to topics authored in XDITA, HDITA, and MDITA. It also links to a topic authored in full DITA 1.3, and provides a key value for the product's name
- An XDITA topic, created by a technical writer, that reuses a `@keyref` from the map and a `@conref` from an MDITA topic
- An HDITA topic, created by a marketing specialist, that reuses a `@keyref` from the map and a `@conref` from an XDITA topic
- An MDITA topic, created by a software developer, that reuses a `@keyref` from the map and a `@conref` from an HDITA topic
4.4.1 Cross-format example: DITA map

The following DITA map links to topics authored in LwDITA’s three formats and DITA 1.3. It also provides a key for the product’s name.

```xml
<map>
  <title>Remote Lighting Setup</title>
  <keydef keys="product-name">
    <topicmeta>
      <keywords>
        <keyword>Remote Network Lighting</keyword>
      </keywords>
    </topicmeta>
  </keydef>
  <topicref href="xdita-topics/bulbs-to-groups.dita" format="dita"/>
  <topicref href="hdita-topics/low-power.html" format="html"/>
  <topicref href="mdita-topics/basic-concepts.md" format="markdown"/>
  <topicref href="external/dita-topics/contact-info.dita" format="dita"/>
</map>
```

Comment by carloseviapuerto on 6 February 2017
This should be a LwDITA map, right? How do we declare keys in an XDITA map?

4.4.2 Cross-format example: XDITA topic

The following XDITA topic reuses a @keyref from the map and a @conref from an MDITA topic.

```xml
<topic id="bulbs-to-groups">
  <title>Programming Light Bulbs to a Lighting Group</title>
  <shortdesc>You can program one or more light bulbs to a lighting group to operate that group with your remote control.</shortdesc>
  <body>
    <section id="context">
      <p>Your <ph keyref="product-name"/> remote control can manage up to 250 network light bulbs on the same lighting network. When you add a light bulb to the network, you can program it to one or more lighting groups.</p>
      <p id="assign-disclaimer">You must assign a light bulb to at least one lighting group to operate that light bulb.</p>
    </section>
    <section id="steps">
      <ol>
        <li><p conref="basic-concepts.md#basic-concepts/power-off" /></li>
        <li><p>Remove any existing light bulb from the light fixture.</p></li>
      </ol>
    </section>
</topic>
```
Install the network light bulb into the light fixture as you would any standard light bulb. Turn power to the light fixture on.  

4.4.3 Cross-format example: HDITA topic

The following HDITA topic reuses a @keyref from the map and a @conref from an XDITA topic.

```html
<!DOCTYPE html>
<title>Low-Power Networking</title>
<article id="low-power">
<h1>Low-Power Networking</h1>
<p data-hd-class="shortdesc">Your <span data-hd-keyref="product-name" /> operates at a low level of networking power but can successfully connect at long distances because they can send information from light bulb to light bulb.</span></p>
<p id="disconnect-warning" data-hd-class="note">Even in low power networks, be sure to disconnect all devices before performing maintenance tasks.</p>
</article>
```

4.4.4 Cross-format example: MDITA topic

The following MDITA extended profile topic reuses a @keyref from the map and a @conref from an HDITA topic.

```---
id: basic-concepts
shortdesc: You can network LED light bulbs together to operate wirelessly from a remote control.
---

# Basic Concepts of Network Lighting

Network light bulbs from your <ph keyref="product-name" /> work with your light fixtures the same way as standard light bulbs. They are different, however, in a couple of ways:

- The lighting element in the light bulb uses energy-efficient LED technology.
- The light bulb includes wireless technology that allows the light bulb to connect to a network and be managed remotely.

Make sure power to the fixture where you are installing the light bulb is turned OFF.</ph>```
5 Lightweight DITA tools

Several tools already exist to support organizations who want to explore using LwDITA.

<table>
<thead>
<tr>
<th>To be covered here:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Plug-in created and maintained by Jarno Elovirta: <a href="https://github.com/jelovirt/dita-ot-markdown">https://github.com/jelovirt/dita-ot-markdown</a></td>
</tr>
<tr>
<td>• Tool in-progress by Tim Grantham for the Lightweight DITA subcommittee</td>
</tr>
<tr>
<td>• oXygen XML Author 18.1</td>
</tr>
<tr>
<td>• CitecFemton</td>
</tr>
<tr>
<td>• DoX CMS</td>
</tr>
</tbody>
</table>

The DITA Technical Committee expects that the release of Lightweight DITA as an OASIS standard will lead to a rapid increase in the number of commercial and open-source tools that provide support for LwDITA.
Appendix A LwDITA elements and attributes

This section lists the elements and attributes that are available in LwDITA.

Appendix A.1 DITA 1.3 elements in LwDITA

This topic lists the DITA 1.3 elements that are available in LwDITA. It also lists how to represent them in XDITA, HDITA, and MDITA.

Comment by Kristen J Eberlein on 21 December 2016

The following table is based on the DTDs in the subcommittee GitHub repository. It does not include `<object>` and `<param>`. Is the plan that they are to be used as the specialization basis for the multimedia elements but NOT to be available in Lightweight DITA?

I also added `<desc>`; content is needed for the HDITA and MDITA equivalents.

<table>
<thead>
<tr>
<th>Component</th>
<th>XDITA</th>
<th>HDITA</th>
<th>MDITA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alternate text</td>
<td><code>&lt;alt&gt;</code></td>
<td>Attribute on <code>&lt;img&gt;</code></td>
<td>[text]</td>
</tr>
<tr>
<td>Body</td>
<td><code>&lt;body&gt;</code></td>
<td><code>&lt;body&gt;</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td>Bold</td>
<td><code>&lt;b&gt;</code></td>
<td><code>&lt;strong&gt;</code></td>
<td>** or __</td>
</tr>
<tr>
<td>Cross reference</td>
<td><code>&lt;xref&gt;</code></td>
<td><code>&lt;href&gt;</code></td>
<td>[link]/URI &quot;title&quot;</td>
</tr>
<tr>
<td>Data</td>
<td><code>&lt;data&gt;</code></td>
<td><code>&lt;meta&gt;</code></td>
<td>Any variables declared in a YAML front matter header. The front matter must be the first thing in the file and must be set between triple-dashed lines. (MDITA extended profile)</td>
</tr>
<tr>
<td>Definition description</td>
<td><code>&lt;dd&gt;</code></td>
<td><code>&lt;dd&gt;</code></td>
<td><code>&lt;dd&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Definition list entry</td>
<td><code>&lt;dlentry&gt;</code></td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Component</td>
<td>XDITA</td>
<td>HDITA</td>
<td>MDITA</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Definition term</td>
<td><code>&lt;dt&gt;</code></td>
<td><code>&lt;dt&gt;</code></td>
<td><code>&lt;dt&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Definition list</td>
<td><code>&lt;dl&gt;</code></td>
<td><code>&lt;dl&gt;</code></td>
<td><code>&lt;dl&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Description</td>
<td><code>&lt;desc&gt;</code></td>
<td><code>&lt;caption&gt;</code> in <code>&lt;table&gt;</code>; <code>&lt;figcaption&gt;</code> in <code>&lt;figure&gt;</code>; not applicable in links</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Figure</td>
<td><code>&lt;fig&gt;</code></td>
<td><code>&lt;figure&gt;</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td>Footnote</td>
<td><code>&lt;fn&gt;</code></td>
<td><code>@data-hd-class=&quot;fn&quot;</code></td>
<td><code>&lt;fn&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Comment by Carlos Evia on 21 February 2017 Should this be a custom HDITA element or should we just send people to <a href="https://www.w3.org/TR/html5/common-idioms.html#footnotes">https://www.w3.org/TR/html5/common-idioms.html#footnotes</a></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Comment by carloseviapuerto Should we ask people to use raw XDITA element or implement the Pandoc approach to footnotes? <a href="http://pandoc.org/MANUAL.html#footnotes">http://pandoc.org/MANUAL.html#footnotes</a></td>
</tr>
<tr>
<td>Image</td>
<td><code>&lt;image&gt;</code></td>
<td><code>&lt;img&gt;</code></td>
<td><img src="images/image_name.jpg" alt="alt text for an image" /></td>
</tr>
<tr>
<td>Italics</td>
<td><code>&lt;i&gt;</code></td>
<td><code>&lt;em&gt;</code></td>
<td>* or _</td>
</tr>
<tr>
<td>List item</td>
<td><code>&lt;li&gt;</code></td>
<td><code>&lt;li&gt;</code></td>
<td><code>-</code>, <code>+</code>, or <code>*</code> for ul, and 0-9 and . or ) for ol</td>
</tr>
<tr>
<td>Map</td>
<td><code>&lt;map&gt;</code></td>
<td><code>&lt;nav&gt;</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td>Note</td>
<td><code>&lt;note&gt;</code></td>
<td><code>&lt;p data-hd-class=&quot;note&quot;&gt;</code></td>
<td><code>&lt;note&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Ordered list</td>
<td><code>&lt;ol&gt;</code></td>
<td><code>&lt;ol&gt;</code></td>
<td>See list item</td>
</tr>
<tr>
<td>Paragraph</td>
<td><code>&lt;p&gt;</code></td>
<td><code>&lt;p&gt;</code></td>
<td>Two carriage returns</td>
</tr>
<tr>
<td>Navigation title</td>
<td><code>&lt;navtitle&gt;</code></td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Component</td>
<td>XDITA</td>
<td>HDITA</td>
<td>MDITA</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------</td>
<td>---------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Phrase</td>
<td><code>&lt;ph&gt;</code></td>
<td><code>&lt;span&gt;</code></td>
<td><code>&lt;ph&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Preformatted text</td>
<td><code>&lt;pre&gt;</code></td>
<td><code>&lt;pre&gt;</code></td>
<td><code>text</code></td>
</tr>
<tr>
<td>Prolog</td>
<td><code>&lt;prolog&gt;</code></td>
<td><code>&lt;meta&gt;</code></td>
<td>Provided in YAML header</td>
</tr>
<tr>
<td>Section</td>
<td><code>&lt;section&gt;</code></td>
<td><code>&lt;section&gt;</code></td>
<td><code>##</code> or <code>-----</code> underline</td>
</tr>
<tr>
<td>Short description</td>
<td><code>&lt;shortdesc&gt;</code></td>
<td><code>&lt;p data-hd-class=&quot;shortdesc&quot;&gt;</code></td>
<td>Variable shortdesc in YAML header (MDITA extended profile)</td>
</tr>
<tr>
<td>Table</td>
<td><code>&lt;simpletable&gt;</code></td>
<td><code>&lt;table&gt;</code></td>
<td>Tables in MDITA follow the GitHub Flavored Markdown syntax: &quot;You can create tables with pipes</td>
</tr>
<tr>
<td>Simple table entry</td>
<td><code>&lt;stentry&gt;</code></td>
<td><code>&lt;td&gt;</code></td>
<td>See Table</td>
</tr>
<tr>
<td>Simple table header</td>
<td><code>&lt;sthead&gt;</code></td>
<td><code>&lt;th&gt;</code></td>
<td>See Table</td>
</tr>
<tr>
<td>Simple table row</td>
<td><code>&lt;strow&gt;</code></td>
<td><code>&lt;tr&gt;</code></td>
<td>See Table</td>
</tr>
<tr>
<td>Subscript</td>
<td><code>&lt;sub&gt;</code></td>
<td><code>&lt;sub&gt;</code></td>
<td><code>&lt;sub&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Superscript</td>
<td><code>&lt;sup&gt;</code></td>
<td><code>&lt;sup&gt;</code></td>
<td><code>&lt;sup&gt;</code> in XDITA syntax (MDITA extended profile)</td>
</tr>
<tr>
<td>Title</td>
<td><code>&lt;title&gt;</code></td>
<td><code>&lt;h1&gt;</code> for topic <code>&lt;h2&gt;</code> for section</td>
<td><code>#</code> or <code>===</code> underline for topic <code>##</code> or <code>-----</code> underline for section</td>
</tr>
<tr>
<td>Topic</td>
<td><code>&lt;topic&gt;</code></td>
<td><code>&lt;article&gt;</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td>Topic metadata</td>
<td><code>&lt;topicmeta&gt;</code></td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Topic reference</td>
<td><code>&lt;topicref&gt;</code></td>
<td><code>&lt;href&gt;</code> inside a <code>&lt;li&gt;</code></td>
<td><a href="/URI" title="title">link</a> inside a list item</td>
</tr>
</tbody>
</table>
### Component

<table>
<thead>
<tr>
<th>Component</th>
<th>XDITA</th>
<th>HDITA</th>
<th>MDITA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Underline</td>
<td><code>&lt;u&gt;</code></td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Unordered list</td>
<td><code>&lt;ul&gt;</code></td>
<td><code>&lt;ul&gt;</code></td>
<td>See List item</td>
</tr>
</tbody>
</table>

### Appendix A.2 New elements

This topic lists the new XML elements that are part of LwDITA and how to represent them in XDITA and HDITA. These new elements are not available in the MDITA core profile and, if needed, can be represented with their raw XDITA equivalents as part of the MDITA extended profile.

**Comment by Kristen J Eberlein on 21 December 2016**

Content needed for some rows.

<table>
<thead>
<tr>
<th>Component</th>
<th>XDITA</th>
<th>HDITA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio</td>
<td><code>&lt;audio&gt;</code></td>
<td></td>
</tr>
<tr>
<td>Controls</td>
<td><code>&lt;controls&gt;</code></td>
<td><code>@controls in &lt;audio&gt;</code> or <code>&lt;video&gt;</code></td>
</tr>
<tr>
<td>Fallback</td>
<td><code>&lt;fallback&gt;</code></td>
<td><code>&lt;source&gt;</code></td>
</tr>
<tr>
<td>Footnote reference</td>
<td><code>&lt;fnref&gt;</code></td>
<td><code>??</code></td>
</tr>
</tbody>
</table>

**Comment by carloseviapuerto on 21 February 2017**

This will change once we decide how to map `<fn>` in previous table.

<table>
<thead>
<tr>
<th>Component</th>
<th>XDITA</th>
<th>HDITA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poster</td>
<td><code>&lt;poster&gt;</code></td>
<td><code>@poster in &lt;video&gt;</code></td>
</tr>
<tr>
<td>Source</td>
<td><code>&lt;source&gt;</code></td>
<td><code>&lt;source&gt;</code></td>
</tr>
<tr>
<td>Track</td>
<td><code>&lt;track&gt;</code></td>
<td><code>@track in &lt;audio&gt;</code> or <code>&lt;video&gt;</code></td>
</tr>
<tr>
<td>Video</td>
<td><code>&lt;video&gt;</code></td>
<td><code>&lt;video&gt;</code></td>
</tr>
</tbody>
</table>

### Appendix A.3 DITA 1.3 attributes in LwDITA

This topic lists the DITA 1.3 attributes that are available in LwDITA. It also lists how to represent them in XDITA and HDITA. These attributes are not available in the MDITA core profile and, if needed, can be represented with their raw XDITA equivalents as part of the MDITA extended profile.

**Comment by Kristen J Eberlein on 29 December 2016**

The following table is based on the DTDs in the subcommittee GitHub repository.
Content is needed for the HDITA and MDITA equivalents.

Comment by Carlos Evia on 5 January 2017
We need to be very careful when proposing attributes for MDITA. Attribute-heavy Markdown is sacrilegious.

<table>
<thead>
<tr>
<th>Component</th>
<th>XDITA</th>
<th>HDITA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content reference</td>
<td>@conref</td>
<td>@data-hd-conref</td>
</tr>
<tr>
<td>Direction</td>
<td>@dir</td>
<td>@dir</td>
</tr>
<tr>
<td>Expanse</td>
<td>@expanse</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Frame</td>
<td>@frame</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Identifier</td>
<td>@id</td>
<td>@id</td>
</tr>
<tr>
<td>Importance</td>
<td>@importance</td>
<td>@data-hd-importance</td>
</tr>
<tr>
<td>Key reference</td>
<td>@keyref</td>
<td>@data-hd-keyref</td>
</tr>
<tr>
<td>Language</td>
<td>@xml:lang</td>
<td>@lang</td>
</tr>
<tr>
<td>Output class</td>
<td>@outputclass</td>
<td>@class</td>
</tr>
<tr>
<td>Props</td>
<td>@props</td>
<td>@data-hd-props</td>
</tr>
<tr>
<td>Scale</td>
<td>@scale</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Translate</td>
<td>@translate</td>
<td>@translate</td>
</tr>
<tr>
<td>Type</td>
<td>@type</td>
<td>@data-hd-type</td>
</tr>
</tbody>
</table>
Appendix B Acknowledgments

The following individuals participated in the creation of this document and are gratefully acknowledged.

- Robert D. Anderson, IBM
- Jan Benedictus, Fonto Group BV
- Deb Bissantz, Healthwise
- Stan Doherty, Individual member
- Kristen James Eberlein, Eberlein Consulting LLC
- Nancy Harrison, Individual member
- Carlos Evia, Virginia Tech
- Mark Giffin, Individual member
- Tim Grantham, Precision Content Authoring Solutions Inc.
- Ullakaisa Kalander, Citec
- Eliot Kimber, Individual member
- Keith Schengli-Roberts, IXIASOFT
- Michael Priestley, IBM

In addition, the OASIS DITA Technical Committee also would like to recognize the following people for their insights and support:

- Jarno Elovirta
- Kevin Lewis
Appendix C Revision history

The following table contains information about revisions to this document.

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Description of changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>5 November 2016</td>
<td>Carlos Evia</td>
<td>Created stub files for working draft.</td>
</tr>
<tr>
<td>02</td>
<td>17 November 2016</td>
<td>Carlos Evia</td>
<td>Updates to structure of working draft.</td>
</tr>
<tr>
<td>03</td>
<td>21 and 22 November 2016</td>
<td>Carlos Evia</td>
<td>Added content, examples, and information about YAML.</td>
</tr>
</tbody>
</table>
| 04       | 24 November 2016      | Kristen James Eberlein              | • Added table to revision history  
• Corrected front matter metadata  
• Added terminology topic  
• General edits for clarity and adherence to IBM style  
• Reworked XDITA, HDITA, and MDITA topic clusters |
| 05       | 30 November 2016      | Kristen James Eberlein              | • Updated cover page metadata to match OASIS template  
• General edits for clarity and adherence to IBM style  
• General reorganization |
| 06       | 01 December 2016      | Kristen James Eberlein              | • Added content to Appendix A  
• Added draft comments throughout  
• Added items to "References" topic |
| 07       | 01 December 2016      | Kristen James Eberlein              | • Added initial content to "Stricter content model" topic  
• Added initial content to "Subset of reuse mechanisms" topic |
<p>| 08       | 05 December 2016      | Carlos Evia                         | Added initial examples for maps in XDITA, HDITA, and MDITA                              |
| 09       | 06 December 2016      | Kristen James Eberlein              | Generated working draft 01                                                             |
| 10       | 15 December 2016      | Kristen James Eberlein              | • Replaced authoring mode with authoring format                                           |</p>
<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Description of changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>15 December 2016</td>
<td>Carlos Evia</td>
<td>• Updated chapter three: &quot;Lightweight DITA design&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added xref for Evia-Priestley article</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Cleaned some sections that implied XDITA equals LwDITA</td>
</tr>
<tr>
<td>12</td>
<td>15 December 2016</td>
<td>Carlos Evia</td>
<td>Improved consistency of LwDITA acronym usage across topics and chapters</td>
</tr>
<tr>
<td>13</td>
<td>16 December 2016</td>
<td>Carlos Evia</td>
<td>Cleaned some short descriptions and added content to authoring formats</td>
</tr>
<tr>
<td>14</td>
<td>16 December 2016</td>
<td>Kristen James Eberlein</td>
<td>• Updated &quot;Acknowledgements&quot; to include everyone who attended the call in 14 December 2016</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated audience for HDITA</td>
</tr>
<tr>
<td>15</td>
<td>21 December 2016</td>
<td>Kristen James Eberlein</td>
<td>• Updated references to list blogs and PDFs of conference session slide decks</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated &quot;Subset of DITA 1.3 elements&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated appendix A, &quot;Lightweight DITA elements and attributes&quot;</td>
</tr>
<tr>
<td>16</td>
<td>29 December 2016</td>
<td>Kristen James Eberlein</td>
<td>• Edits to appendix A</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Generated working draft #2</td>
</tr>
<tr>
<td>17</td>
<td>30 December 2017</td>
<td>Kristen James Eberlein</td>
<td>Response to feedback from Deb Bissantz, Healthwise:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added &lt;u&gt; to 3.2, &quot;Stricter content model&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Changed 2.1, &quot;Simplified model&quot; to &quot;Simplified structure&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Edited 4.2.1, &quot;Audience for HDITA&quot;</td>
</tr>
<tr>
<td>18</td>
<td>5 January 2017</td>
<td>Carlos Evia</td>
<td>• Commented out the specialization section from map (it will be the main topic</td>
</tr>
<tr>
<td>Revision</td>
<td>Date</td>
<td>Editor</td>
<td>Description of changes</td>
</tr>
<tr>
<td>----------</td>
<td>--------------</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>of a future committee note or publication)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated tables of elements and attributes to reflect input from LwDITA's syntax taskforce</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Commented out mentions of the specialization mechanism as an initial LwDITA component.</td>
</tr>
<tr>
<td>19</td>
<td>9 January 2017</td>
<td>Kristen James Eberlein</td>
<td>• Corrected date in revision history</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added back specialization topics</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Corrected typos in draft comments; removed outdated draft comments</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated acknowledgements to include Deb Bissantz, Healthwise</td>
</tr>
<tr>
<td>20</td>
<td>9 January 2017</td>
<td>Kristen James Eberlein</td>
<td>Response to feedback from Ulla Kalandar, Citec:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added <code>&lt;note&gt;</code> to 3.1, &quot;Subset of DITA 1.3 elements&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added draft comments with feedback from Citec</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Added <code>@type</code> to Appendix A.3, &quot;DITA 1.3 attributes in Lightweight DITA&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated acknowledgements to include Ulla Kalandar, Citec</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated &quot;Notices&quot; to have a copyright date of 2017</td>
</tr>
<tr>
<td>21</td>
<td>9 January 2017</td>
<td>Carlos Evia</td>
<td>Added raw draft paragraphs to capture feedback from SC's meeting and minutes of note call from December.</td>
</tr>
<tr>
<td>22</td>
<td>17 January 2017</td>
<td>Carlos Evia</td>
<td>• Added content for multimedia elements</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Improved LwDITA map topic</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Standardized use of LwDITA in document</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Drafted rationale for specialization mechanism</td>
</tr>
<tr>
<td>Revision</td>
<td>Date</td>
<td>Editor</td>
<td>Description of changes</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------</td>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>23</td>
<td>19 January 2017</td>
<td>Carlos Evia</td>
<td>Added content about compatibility of authoring modes.</td>
</tr>
<tr>
<td>24</td>
<td>23 January 2017</td>
<td>Carlos Evia</td>
<td>Updated attributes and elements appendixes</td>
</tr>
<tr>
<td>25</td>
<td>23 January 2017</td>
<td>Kristen James Eberlein</td>
<td>Generated working draft #3</td>
</tr>
<tr>
<td>26</td>
<td>23 January 2017</td>
<td>Carlos Evia</td>
<td>Added content explaining multimedia elements</td>
</tr>
<tr>
<td>27</td>
<td>24 January 2017</td>
<td>Carlos Evia</td>
<td>Updated content model example to follow note's theme of light bulbs</td>
</tr>
</tbody>
</table>
| 28       | 24 January 2017    | Carlos Evia                 | • Cleaned multimedia elements topic  
• Added stubs for new 4.4 section on cross-format LwDITA content and examples             |
<p>| 29       | 25 January 2017    | Carlos Evia                 | Added draft of cross-format topics                                                     |
| 30       | 25 January 2017    | Carlos Evia                 | Updated the cross-format content example                                               |
| 31       | 30 January 2017    | Carlos Evia                 | Added specialization content from Mark G and Michael P                                  |
| 32       | 30 January 2017    | Carlos Evia                 | Incorporated feedback from John H                                                      |
| 33       | 30 January 2017    | Carlos Evia                 | Edits after reading most recent draft                                                   |
| 34       | 30 January 2017    | Carlos Evia                 | Improved consistency in tables of elements and attributes                              |
| 35       | 30 January 2017    | Carlos Evia                 | Mapped &lt;desc&gt; to HDITA (not applicable in MDITA)                                       |
| 36       | 31 January 2017    | Carlos Evia                 | Cleaned specialization introductory topic                                               |
| 37       | 1 February 2017     | Carlos Evia                 | Added specialization example                                                           |
| 38       | 1 February 2017     | Carlos Evia                 | Added Tim G to contributors and incorporated Tim's content on specialization            |</p>
<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Editor</th>
<th>Description of changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>3 February 2017</td>
<td>Carlos Evia</td>
<td>Incorporated feedback from Ullakaisa K</td>
</tr>
<tr>
<td>40</td>
<td>6 February 2007</td>
<td>Carlos Evia</td>
<td>Added fnref to new elements table and moved all new elements to the same table</td>
</tr>
<tr>
<td>41</td>
<td>6 February 2017</td>
<td>Carlos Evia</td>
<td>Added map elements to table of LwDITA elements</td>
</tr>
<tr>
<td>42</td>
<td>6 February 2017</td>
<td>Carlos Evia</td>
<td>Removed all references to the proposed specialization mechanism after SC's Feb 6, 2017 call</td>
</tr>
<tr>
<td>43</td>
<td>7 February 2017</td>
<td>Carlos Evia</td>
<td>Changed MDITA's dependence on raw HTML to raw XML for structural elements and reuse</td>
</tr>
<tr>
<td>44</td>
<td>8 February 2017</td>
<td>Carlos Evia</td>
<td>Updated syntax for shortdesc in MDITA topics</td>
</tr>
<tr>
<td>45</td>
<td>9 February 2017</td>
<td>Carlos Evia</td>
<td>Updated Tim G's affiliation in acknowledgments</td>
</tr>
<tr>
<td>46</td>
<td>10 February 2017</td>
<td>Carlos Evia</td>
<td>Incorporated Jan B's content in response to John H's questions. Added Jan B to list of contributors</td>
</tr>
<tr>
<td>47</td>
<td>20 February 2017</td>
<td>Carlos Evia</td>
<td>Added fnref topic and draft comments following 02/20/17 subcommittee meeting</td>
</tr>
<tr>
<td>48</td>
<td>21 February 2017</td>
<td>Carlos Evia</td>
<td>Improved topic for footnotes based on Michael P's email from 02/21/17</td>
</tr>
<tr>
<td>49</td>
<td>8 March 2017</td>
<td>Carlos Evia</td>
<td>Added Jarno to acknowledgements and started separation of MDITA for base and extended models</td>
</tr>
<tr>
<td>50</td>
<td>8 March 2017</td>
<td>Carlos Evia</td>
<td>Separated MDITA in core and extended profiles. Generated Working Draft 09</td>
</tr>
</tbody>
</table>